

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STAVEBNÍ

Katedra ekonomiky a řízení ve stavebnictví

# DIPLOMOVÁ PRÁCE

2017

Bc. Daniel Šabart





## ZADÁNÍ DIPLOMOVÉ PRÁCE

### I. OSOBNÍ A STUDIJNÍ ÚDAJE


Příjmení: Šabart	Jméno: Daniel	Osobní číslo: 395979
Zadávací katedra: k126		
Studijní program: Stavební inženýrství		
Studijní obor: Projektový management a inženýring (P)		

### II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce: Vyhodnocení přínosu nástrojů detekce kolizí s využitím BIM	
Název diplomové práce anglicky: Evaluation of clash detection benefits using BIM tools	
Pokyny pro vypracování: Rámcový obsah diplomové práce - Úvod do problematiky a její vymezení; popis principu detekce kolizí, definice základních pojmů - Rešerše existujících nástrojů detekce kolizí s využitím BIM - Srovnání vybraných nástrojů detekce kolizí s využitím BIM za účelem zjištění jejich základních odlišností ve funkčnosti - Ověření přínosu detekce kolizí s využitím BIM - praktické úlohy - Závěr a vyhodnocení	
Seznam doporučené literatury: Eastman, Ch.: BIM Handbook, 2011. ISBN: 978-0-470-54137-1.	
Jméno vedoucího diplomové práce: Ing. Petr Matějka	
Datum zadání diplomové práce: 7.10.2016	Termín odevzdání diplomové práce: 8.1.2017 <i>Údaj uveďte v souladu s datem v časovém plánu příslušného ak. roku</i>
 Podpis vedoucího práce	 Podpis vedoucího katedry

### III. PŘEVZETÍ ZADÁNÍ

*Beru na vědomí, že jsem povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je nutné uvést v diplomové práci a při citování postupovat v souladu s metodickou příručkou ČVUT „Jak psát vysokoškolské závěrečné práce“ a metodickým pokynem ČVUT „O dodržování etických principů při přípravě vysokoškolských závěrečných prací“.*

7.10.2016 Datum převzetí zadání	 Podpis studenta(ky)
------------------------------------	---



## PROHLÁŠENÍ

Jméno diplomanta: Bc. Daniel Šabart

Název diplomové práce: Vyhodnocení přínosu nástrojů detekce kolizí s využitím BIM

Prohlašuji, že jsem uvedenou diplomovou práci vypracoval samostatně pod vedením školitele Ing. Petra Matějky

Použitou literaturu a další materiály uvádím v seznamu použité literatury.

V Praze dne .....

.....

podpis



# **Vyhodnocení přínosu nástrojů detekce kolizí s využitím BIM**

**Evaluation of clash detection benefits using BIM tools**



## ABSTRAKT

Diplomová práce se zabývá vyhodnocením přínosu použití BIM nástrojů v úlohách detekcí kolizí. V úvodu je čtenáři přiblížena problematika práce s informačními modely. Dále je uvedena koncepce Revize modelu, která obsahuje náměty pro rozvoj použití BIM z hlediska tvorby a analýzy modelů. Následně je popsána definice a rozdělení kolizí. Součástí práce je rešerše dostupných BIM nástrojů na detekci kolizí. Vybrané programy byly analyzovány a jejich možnosti detailně popsány. Na základě získaných poznatků byl vybrán program pro další použití ve výzkumu. Následně byly sestaveny úlohy pro praktické porovnání efektivity detekce kolizí. Ty byly zadány řešitelům a zároveň zpracovány ve vybraném programu. Výsledky byly dosazeny do sestavených rovnic a jako hlavní výstup byla určena relativní účinnost detekce. Práce může sloužit jako podklad pro rozhodování o investici do určitého typu BIM nástrojů.

Klíčová slova: BIM, Revize modelu, kolize, BIM software, účinnost detekce kolizí

## ABSTRACT

The diploma thesis deals with evaluation of benefits using BIM tools in clash detection tasks. The first part is focused on explaining the problematics of work with information models. In the next section, the conception of Model checking is described. It involves topics for possible development of BIM usage in the process of model creation and analysis. Further on, the definition and sorting of clash types is described. This thesis includes a research of available clash detection BIM software. Selected tools were analysed and their capabilities were described in detail. One of the programs has been chosen on the basis of acquired information for further use in the research. Then model tasks for practical comparison of clash detection efficiency were created. These were given to both researchers and software for processing. The outcomes were substituted into compiled equations. As the main result, the relative efficiency of detection has been defined. This thesis can be used to enhance decision making about investment in certain type of BIM tools.

Keywords: BIM, Model checking, clashes, BIM software, efficiency of clash detection



## OBSAH

PROHLÁŠENÍ.....	iii
ABSTRAKT.....	v
ABSTRACT.....	v
OBSAH.....	vi
SEZNAM POUŽITÝCH ZKRATEK .....	viii
ZÁKLADNÍ POJMY .....	ix
ÚVOD .....	1
1 Práce s BIM modelem.....	3
1.1 Informační modelování ve výstavbových projektech .....	3
1.2 Kontroly informačních modelů.....	5
1.2.1 Revize modelu.....	5
1.3 Kolize .....	9
1.3.1 Typy kolizí.....	9
1.3.2 Dopady kolizí.....	16
1.3.3 Řešení kolizí.....	17
2 Softwarové nástroje pro detekci kolizí v BIM modelu.....	20
2.1 Kritéria výběru softwaru .....	20
2.1.1 Stand-alone software.....	20
2.1.2 IFC.....	21
2.2 Přehled vybraných programů.....	23
2.3 Vlastnosti vybraných programů .....	23
2.4 Popis vlastností porovnávaných programů.....	26
2.4.1 Navisworks Manage .....	26
2.4.2 Tekla BIMSight .....	31
2.4.3 Solibri Model Checker .....	34
2.4.4 BIMReview .....	39
2.5 Vyhodnocení vlastností porovnávaných softwarů.....	42
3 Vyhodnocení přínosu využití BIM pro detekci kolizí.....	43
3.1 Typy úloh a způsob řešení .....	43
3.2 Tvorba BIM modelů projektů .....	44
3.3 Zpracování úloh řešiteli .....	45
3.4 Zpracování úloh BIM nástrojem .....	46



4	Srovnání zpracování úloh.....	47
4.1	Relativní účinnost detekce .....	48
4.1.1	Koeficient chybovosti.....	48
4.1.2	Určení průměrného výkonu řešitelů.....	49
4.1.3	Určení průměrného výkonu softwaru .....	51
4.2	Číselné vyhodnocení relativní účinnosti.....	52
	ZÁVĚR.....	55
	Reference.....	58
	Seznam tabulek.....	60
	Seznam obrázků.....	60
	Seznam rovnic.....	61
	Seznam příloh .....	61



## SEZNAM POUŽITÝCH ZKRATEK

<b>BIM</b>	Building Information Modelling
<b>BOZP</b>	Bezpečnost a ochrana zdraví při práci
<b>ISO</b>	International Organization for Standardization, Mezinárodní organizace pro normalizaci*
<b>GIS</b>	Geographic Information System, Geografický Informační Systém
<b>GPS</b>	Global Positioning System, Globální polohový systém*
<b>TZB</b>	Technická zařízení budov
<b>ROI</b>	Return On Investment, Návratnost investice*
<b>IFC</b>	Industry Foundation Classes
<b>IFCZIP</b>	Zippered Industry Foundation Classes, Archivované IFC*
<b>BCF</b>	BIM Collaboration Format

\* volný překlad





## ZÁKLADNÍ POJMY

Facility management = v této práci nese význam profesního odvětví, které si klade za cíl zajištění funkcionality a chodu budovy či zařízení. Zpravidla využívá moderní technologie, procesy, postupy a inovace.

Clash detection = systém vyhledávání konfliktů mezi prvky modelu.

Stand-alone = Nástroj, který není integrován do jiného programu a tvoří tak nezávislý a ucelený software.

Work-flow = obecný výraz pro postup práce (například postup při modelování, navrhování, apod.).

Buffer = označení prostorového obalu kolem určité konstrukce či systému.

Cloud = internetové nebo serverové úložiště používané ke sdílení souborů či projektů.

Add-on = rozšíření programů pro podporu jiných softwarů, zásuvný modul.



## ÚVOD

Rozvoj informačních technologií dal v posledních letech vzniknout systému informačního modelování staveb, známém pod zkratkou BIM (*Building Information Modelling*). Nejen stavební společnosti těží z jeho nástrojů pro zkvalitnění výstavbových procesů. Zájem na použití této metodiky mají například i developerské společnosti. Vhodným implementováním BIM do procesu výstavby a životního cyklu stavby lze dosáhnout značných úspor. Za jeden ze stěžejních prvků metodiky BIM je považován značně automatizovaný systém detekce kolizí. Identifikací konfliktů a jejich včasným eliminováním již ve fázi návrhu lze ušetřit náklady, které by byly nutné pro jejich odstranění v pozdějších etapách výstavby. Pro zavedení procesu kontroly chyb v návrhu projektu je však třeba vynaložit určité náklady a úsilí.

Pro tuto práci byly stanoveny čtyři hlavní cíle. Lze říci, že stěžejním cílem práce je Cíl 4, který je dosažen postupným splňováním Cílů 1,2 a 3. Jednotlivé cíle byly sepsány a seřazeny dle svého významu následovně:

- Cíl 1**      **Detailní popis teorie kolizí a jejích principů**
- Cíl 2**      **Provedení průzkumu dostupnosti BIM nástrojů na detekci kolizí**
- Cíl 3**      **Analýza a podrobný popis vlastností vybraných programů**
- Cíl 4**      **Vyhodnocení přínosu použití BIM nástrojů pro detekci kolizí oproti klasickému postupu**

Klasický postup detekce kolizí v podobě překrývání výkresů a ruční označování konfliktů by mohl být mnohdy považován za dostačující. Výzkum uvedený v této práci zahrnuje srovnání řešení modelových úloh člověkem a softwarem. Společnostem, které uvažují nad zavedením BIM metodiky do procesu kontroly, by mohly výsledky této práce pomoci při rozhodování.

V první části této práce je čtenář seznámen s problematikou práce s informačními modely. Autor se zde zaměřuje na současný stav využívání BIM metodiky a jejích základních myšlenek. Zároveň je v této části uveden návrh, jakým směrem by se mohla v budoucnosti vyvíjet práce s BIM modely z hlediska jejich tvorby a analýzy.

Definici kolizí v BIM modelech a jejich rozdělení je věnována druhá polovina této kapitoly. Dále jsou popisovány možné dopady ignorovaných kolizí. Ty se zpravidla projevují zvýšenými náklady na předělání. Toto tvrzení je potvrzeno citovanými údaji z případových studií, které jsou zaměřeny na průzkum výše úspor při použití BIM ve výstavbových projektech. V další kapitole jsou popisovány typy řešení kolizí i s jejich



úskalími. To je doplněno o praktickou ukázkou autorem zpracovaného postupu řešení i s fotodokumentací ze skutečného projektu, kterého se autor zúčastnil.

V druhé části práce je popsán způsob výběru softwarů v rámci průzkumu dostupných BIM nástrojů na detekce kolizí. Pro jeho účely byla stanovena určitá kritéria a následně sestavena tabulka, obsahující vybrané programy a stručné okomentování jejich vlastností. Tyto programy byly následně podrobeny detailnímu zkoumání. Na základě poznatků, kvality zpracování a v neposlední řadě i subjektivního dojmu autora, byl vybrán software, který byl následně použit pro zpracování modelových úloh.

Další část práce je věnována popisu praktických úloh, použitých při zkoumání přínosu a efektivity použití vybraného BIM nástroje. Autor popisuje zpracování úloh v projekčním programu, přípravu podkladů pro řešitele a důvody jejich charakteristických vlastností. Dále je popsán postup výběru reprezentativního vzorku řešitelů a průběh řešení zadaných úkolů i s jejich výsledky. Jako poslední je uveden proces řešení stejných úloh ve vybraném programu včetně související přípravy softwaru, kontroly a příslušných výstupů.

V poslední části diplomové práce je uveden autorem navržený výpočet, sestávající z obecných rovnic. Cílem je určit číselné vyhodnocení modelových úloh a explicitně tak vyjádřit poměr mezi efektivitou práce člověka a softwaru.



# 1 Práce s BIM modelem

V posledních letech se ve stavebnictví stále více začíná prosazovat projektování staveb za použití informačních modelů. To je obecně známo pod názvem *Building Information Modelling (BIM)*. BIM modelem se rozumí takový digitální model projektu, který obsahuje nejen geometrická data, ale také velké množství doplňujících informací a specifikací, využitelných v různých fázích projektu. Může se tak jednat o podrobnou materiálovou skladbu a cenové údaje jednotlivých konstrukčních prvků, jejich dodavatelé, životnost, návaznosti na další prvky, datum zabudování, přehled plánovaných údržeb a mnoho dalšího.

## 1.1 Informační modelování ve výstavbových projektech

Údaje obsažené v BIM modelu lze využít v mnoha odvětvích stavební produkce, projektového managementu nebo facility managementu a zpravidla je tak činěno s příslibem ušetření nákladů, které by bez použití modelu a dat z něj plynoucích mohly s velkou pravděpodobností narůstat.

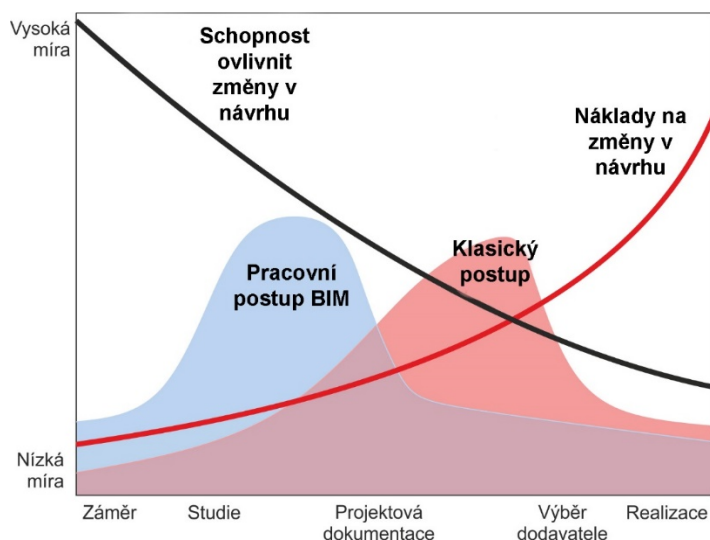
V přípravných fázích projektu mohou takové informace sloužit pro optimalizaci návrhu a snížení budoucích nákladů na odstranění nesrovnalostí způsobených špatnou koordinací nebo i pouhým lidským faktorem. V realizační fázi lze těžit z informačního modelu z hlediska monitorování a kontroly provádění stavby, koordinování dodavatelů nebo jako částečný dohled nad dodržováním BOZP. V provozní fázi jsou data z modelu využívána při správě budovy či objektu ve smyslu kontroly údržby, energetické náročnosti, zastarávání a celkové funkčnosti.

I přes vyšší počáteční náklady je BIM stále více využíván projektanty, což vede k modernizaci softwarových nástrojů a vývoji aplikací s konkrétními BIM specializacemi. Metodika BIM se za poslední téměř dvě dekády dostala na výsluní po celém světě. Za světové lídry lze považovat Finsko, Singapur, Velkou Británii nebo Spojené státy americké [1]. Ve většině z těchto zemí je BIM nedílnou součástí projektů veřejného sektoru. Od loňského roku je ve Velké Británii budoucí realizace funkčního informačního modelu podmínkou pro zařazení do výběrového řízení veřejných zakázek. V Singapuru byl například zaveden systém digitálního schvalování projektu ve stavebním řízení *CoreNet e-Submission System* [2]. Tento postup se osvědčil a tamní projekty tak mohou těžit z časové úspory již v přípravné fázi.

I přes všechny potenciální výhody a přínosy není metodika BIM bez nedostatků. Jednou z překážek je vysoká počáteční investice. Z hlediska méně znalého investora se může projektování v BIM zdát jako přežitek a zbytečné navýšení nákladů činnosti, která zdánlivě nedává smysl a nepřináší užitek. V České republice je BIM teprve v prvních

fázích potřebné implementace. Obecně lze říci, že českému investorovi je třeba pravý přínos BIM vysvětlit a přesvědčit ho o jeho kvalitách. Někteří investoři pak vyžadují BIM, aniž by přesně věděli, o co se vlastně jedná a co mají požadovat. Zároveň mívají nereálné požadavky vlivem marketingu, prosazující BIM.

Vytvoření informačního modelu je poměrně komplikovaný proces, na kterém se podílí více osob. Zejména u projektantů je pak důležitá kooperace a nezbytné znalosti. To se nevyhnutelně odráží na pořizovací ceně modelu. Investor potom musí pochopit, že při správném použití BIM modelu lze dosáhnout úspor ve všech etapách projektu, především pak ve fázi užívání. Průběh uvažovaného přínosu metodiky je zobrazen v diagramu (obrázek 1). V této chvíli se však nabízí otázka, pro jak velký projekt převažují potenciální úspory nad náklady na vytvoření BIM modelu. Je tak na každém investoru, aby z rozmyslem zvážil použití zmíněné metodiky a vyhodnotil tak, zda pro něj je či není ekonomicky výhodná.



**Obrázek 1: Náklady na změny projektu**

Zdroj: vlastní s využitím grafu [3], původní autor Patrick Macleamy

Ve veřejném sektoru je postup zavádění BIM odlišný. Jak již bylo zmíněno, státy s pokročilým užíváním BIM jej často zahrnuly do podmínek pro výběrové řízení při zadávání veřejných zakázek. K tomuto kroku je však potřebná platná legislativa, která v České republice prozatím chybí. Usnesení vlády České republiky č. 958 ze dne 2. 11. 2016 je příslibem zahájení zavádění BIM v legislativní podobě. [4] V současnosti je však problémem absence standardů, vyhlášek či předpisů, které by měly definovat a unifikovat postupy tvorby BIM modelů. To zahrnuje mimo jiné sestavení knihoven konstrukčních prvků a jejich obecné definice, kterými by se projektanti měli řídit. V současné době však existují tendence stavebních firem podílet se na tvorbě standardů. To by mohlo představovat způsob přípravy materiálů pro legislativní předpisy. Je ovšem



zapotřebí kontroly pro zajištění objektivního postupu tak, aby nebyly výrazně znevýhodněny společnosti, nepodílející se na tvorbě podkladů.

## 1.2 Kontroly informačních modelů

S neustále narůstající komplexností stavebních projektů stoupá i nutnost kvalitní a spolehlivé kontroly veškerých údajů, které obsahují. Až 40% všech nedostatků a defektů může souviset s chybami v přípravné fázi projektu [5]. Je ovšem důležité si uvědomit, že i dobře sestavený, automatizovaný postup musí být v odpovídajícím rozsahu doplněn lidským faktorem. Jako příklad lze uvést počáteční určení nejdůležitějších prvků. Záleží na požadované podrobnosti. Nicméně stále platí, že pro nejpodstatnější prvky dává kontrola největší smysl a přínos. Určení hlavních předmětů kontroly se provádí například pomocí *Paretova pravidla (80/20)*, kdy 20% prvků představuje 80% celkových nákladů.

I přes to, že se ve většině případů nelze plně spolehnout na automaticky generované výsledky bez následného posouzení a vyhodnocení odborníkem, skýtá takový postup značný potenciál a při vhodném využití dokáže přinést snížení rizik či finanční a časové úspory. Samotná kontrola a detailní analýza návrhu se dá považovat za jeden z největších přínosů práce s BIM. Používáním analytických nástrojů je možné dosáhnout vyšší efektivity nejenom při navrhování, ale zejména pak při koordinaci prací a celkovém managementu projektu. Je však na místě zmínit, že při špatném přístupu, vyhodnocení výsledků a nepřiměřené kontrole, může dojít k potlačení inovací nebo rozvoje nových řešení či technologií.

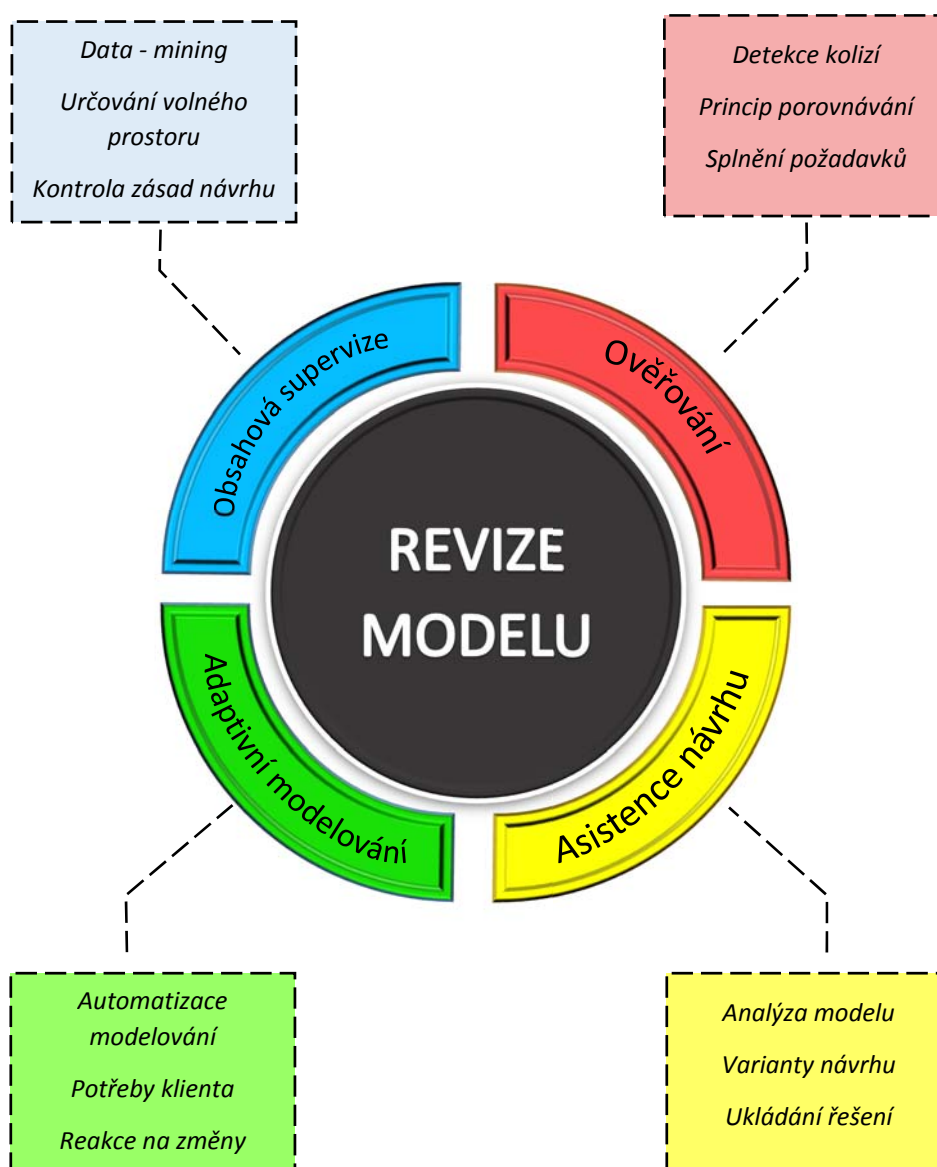
### 1.2.1 Revize modelu

Souhrnnou činnost kontroly modelu lze obecně nazvat *Revize modelu*. Do tohoto postupu je možné zahrnout mimo jiné porovnávání vlastností, atributů a především návazností konstrukčních i doplňkových prvků modelu. Pod tímto pojmem si lze v praxi představit například kontrolu křížení konstrukčních prvků nebo chyby v jejich napojení. Součástí je však i dohled nad dodržováním určitých zásad a předpisů. Obecně je možné definovat chování *Revize modelu* a upravovat postup dle konkrétních požadavků na výstup. V zahraničí se pro podobnou činnost používá pojem *Model checking*. I přes svou značnou významnost však není plně zakotven ve stavebních standardech *ISO* [5].

Celková koncepce *Revize modelu* není v současné době stanovena a nabízí tak prostor pro zavádění nových pojmů a metodik, které se mohou postupem času stát součástí analýzy modelu v rámci kontrolní činnosti stavebních firem, investorů, developerů nebo stavebních úřadů v České republice.

Jedním z návrhů je rozdělení na čtyři základní systémy Revize modelu, kde každý odráží jistý záměr manipulace s informačními modely a představují tak určité náměty, jakým směrem by se mohl koncept Revize modelu ubírat [5]:

- Ověřování
- Asistence návrhu
- Adaptivní modelování
- Obsahová supervize



Obrázek 2: Schéma konceptu Revize modelu

Zdroj: vlastní tvorba



### 1.2.1.1 Ověřování

Záměrem prvního uvedeného systému analýzy modelu je určit, zda je jeho obsah v souladu s určitými požadavky. Ty mohou být určeny například stavebními standardy, vyhláškami, předpisy či regulativy. Výstupem této činnosti je odpověď na definované dotazy ve formě ANO/NE. Základním předpokladem pro úspěšné ověření je určení, které prvky jsou porovnávány a s čím. Nejčastějším případem ověřováním je tzv. *Detekce kolizí* (z anglického *clash detection*), která je podrobně rozebrána v samostatné kapitole této práce (kapitola 1.3). Mimo detekci kolizí spadá do kategorie ověřování již zmíněná kontrola modelu podle pravidel stavebního úřadu pro udělení stavebního povolení. Pravidla pro kontrolu modelů jsou často agregována do sestav, které lze mezi sebou různě kombinovat a analyzovat tak více částí modelu najednou.

### 1.2.1.2 Asistence návrhu

Smyslem druhého systému práce s informačním modelem je napomáhat při návrhu stavby a to obzvláště v oblastech projektu, které jsou obecně problematické nebo se svou specializací a komplikovaností vymykají běžnému projektování. Nejdříve je uživatelem definováno místo výskytu problému a následně programem poskytnut seznam možných akcí k vyřešení. BIM model je tak analyzován a jako výsledek je předloženo více variant určitých konstrukčních nebo technologických řešení, ze kterých může projektant vybírat a tím zlepšovat celkový návrh. K navrhovaným řešením lze přikládat informace, obsahující různé inovativní nápady, kalkulace, poznámky, zkušenosti z praxe a názorné příklady.

Výzvou ve vývoji tohoto systému je pak zavedení metodiky, která bude zajišťovat zapamatování nejlepších, či nejčastěji přijímaných řešení a jejich výraznější doporučení v dalších projektech. To s sebou však přináší potřebu dohledu a ověřování ukládaných řešení za účelem minimalizace nežádoucích komplikací vlivem uložení a následným použitím nesprávných návrhů.

### 1.2.1.3 Adaptivní modelování

Cílem adaptivního modelování je zautomatizování procesu úpravy vlastností prvku s ohledem na prostředí nebo vnější faktory, kterými je ovlivněn. Jako příklad lze uvést myšlenku, kdy se při zvýšení zatížení na sloup zvětší jeho průřez, nebo se, podle předem stanovených pravidel, ponechají stávající rozměry, ale zvýší se počet výztuže. Zavedením adaptivního modelování lze sledovat zefektivnění projektování a minimalizaci chyb, způsobených změnami návrhu nebo špatnou koordinací profesí, které se v modelu objevují. Problematickou částí však může být zajištění změny všech relevantních prvků, nehledě na jejich samotné určení.





Další možnost využití se skrývá i v oblasti komunikace s klienty. Systém přizpůsobování prvků, klientem zadaným podmínkám, nabízí zajímavou možnost rychlého návrhu budovy, sloužícímu jako podklad k diskusi. Společnost *Selvaag Bluethink* vyvinula softwarový nástroj *House Designer*, který využívá podobný systém adaptivních prvků pro vytvoření poměrně detailního, ač stále prvotního návrhu budovy. [5]

#### 1.2.1.4 Obsahová supervize

Protože obsah informačního modelu obecně zahrnuje velké množství druhů údajů, je i jejich celkový management disciplínou, kterou lze podpořit v rámci Revize modelu. Účelem obsahové supervize není sestavování pravidel pro detailní analýzu modelu, nýbrž filtrování specifických informací, které mohou být následně využity jinými programy nebo osobami v průběhu projektu. V dnešní době jsou hojně používány samostatné softwarové nástroje pro prohlížení informačních modelů staveb právě za účelem extrakce konkrétních informací. Mezi ty mohou patřit časové plány, výkazy výměr, cenové údaje, teplotní součinitele prvků, zvukově izolační vlastnosti, požární odolnost, údaje o dodavateli, zárukách, plánu údržeb a oprav a mnoho dalšího. V rámci kontroly modelu se pak nabízí možnost, zaměřit se na tyto informace a porovnávat je s různými předpisy (všechny prvky musí obsahovat určité vlastnosti – viz výše, atd.). Pomocí analýzy časových plánů je pak možné sledovat průběh výstavby a za pomoci aktualizací a predikce modelu pak efektivněji koordinovat práce na staveništi.

Do tohoto systému lze také zahrnout funkci na ohraničení prázdného prostoru v modelu, které bude v budoucnu obsahovat zařízení. V případě, že je například prováděna kontrola detekce kolizí, může dojít k situaci, že se na inkriminované místo v modelu pohlíží jako na volný prostor a neuvažuje se tak jeho budoucí využití.

Modely a jejich obsahy je možné na obecné úrovni porovnávat také mezi sebou. Lze uvést příklad, kdy má developerská společnost určité požadavky na rozměry obytných (nebo i jiných) místností, míru prosklení fasády nebo dalších parametrů a je potřeba zkontrolovat nově sestavený návrh pomocí modelu na dříve zpracovaný projekt. Porovnáním určitých typů podmínek lze dosáhnout unifikovaných řešení u všech projektů a dostat tak požadovaným zásadám.



## 1.3 Kolize

Původní myšlenka detekce prostorových konfliktů mezi objekty se objevila v odvětví herního průmyslu. Při programování prvních počítačových her se zobrazením ve třech dimenzích narazili vývojáři na doposud neznámý problém. Během modelování tras objemových prvků zjistili, že se nevyhnou jejich konfliktu a musí tak začít řešit, co se stane, když se dvě entity dotknou. V praxi to znamenalo vyvinout novou část zdrojového kódu, která měla obecně definovat chování všech inkriminovaných objektů a předejít tak problémům, jako například procházení virtuálních osob zdmi, propadávání předmětů pevnou podlahou a mnoho dalších. Základním postupem bylo zkoumání, zda mají dva objekty stejné souřadnice na osách x, y a z, tedy že se nacházejí na stejném místě v prostoru. Vzhledem k dynamičnosti her bylo také třeba uvažovat obsazení oněch souřadnic i z hlediska času. Řešením bylo zavedení určitých vlastností prvku a sestavení pravidel jeho chování, pokud systém detekuje prostorový konflikt s jiným objektem.

S rozvojem výpočetní techniky se začalo tímto problémem zabývat i stavebnictví prostřednictvím informačních modelů staveb. Do jisté míry byly převzaty programovací postupy na automatické vyhledávání kolizí a označování problematických prvků. Postupem času se systém detekce kolizí v BIM neustále zlepšoval. V současnosti pokračuje výzkum a vývoj nástrojů, čímž se stávají stále sofistikovanějšími pomocníky při práci s informačními modely staveb a návrhu konstrukcí obecně.

Je však třeba zdůraznit, že systém kontroly kolizí je stále jen nástrojem managementu projektu, který dohlíží na správný návrh a přípravu stavby. Není zárukou správného provedení a nenahrazuje tak odborný dohled na stavbách. I když existuje mnoho ohlasů, které hovoří o zefektivnění stavebních prací s použitím BIM modelu, vždy je třeba kontrolovat a instruovat stavební výrobu in situ. Jedině pak lze oprávněně požadovat určitý přínos informačního modelování a s ním spojené detekce kolizí.

### 1.3.1 Typy kolizí

Jak již bylo zmíněno v předchozí kapitole, kolize nastává v případě, že se dva nebo více objektů nachází ve shodném čase na stejném místě. Problematickou částí návrhu je tedy rozmístění prvků a rozvržení výstavby. Z údajů zahrnutých v modelu BIM se tedy kolizí týká především geometrických a časových informací. Obecně lze rozdělit kolize na následující druhy:

- Tvrdé kolize
- Měkké kolize
- 4D kolize [6] a [7]
- Kolize animovaných předmětů



### 1.3.1.1 Tvrdé kolize

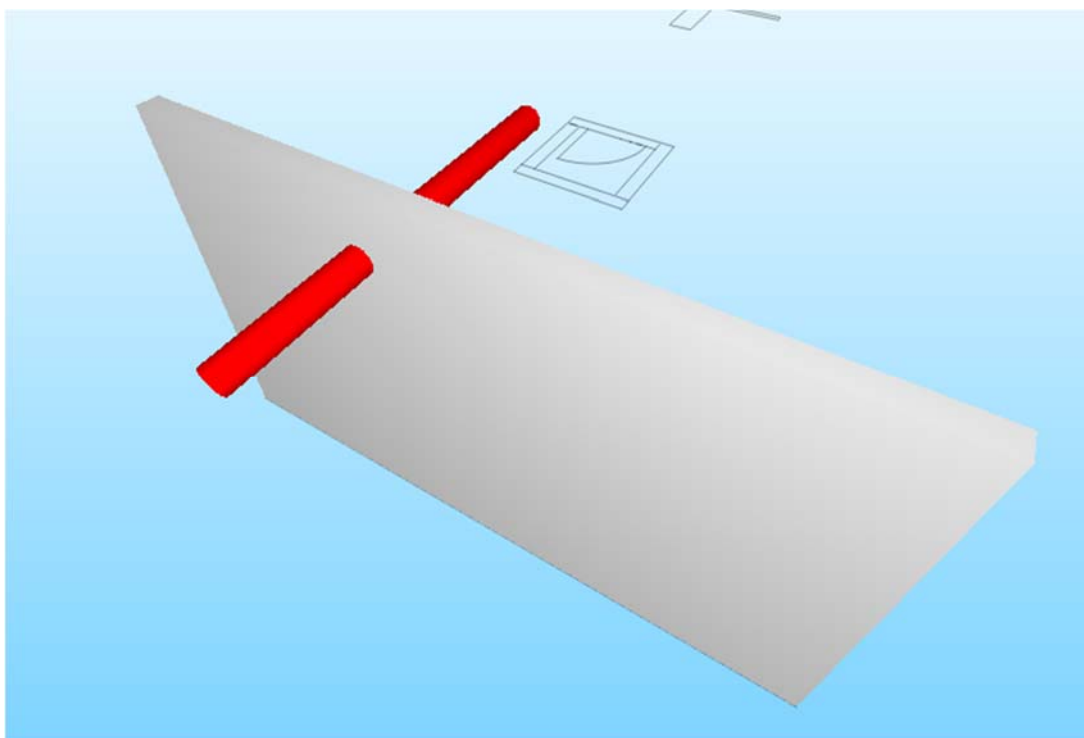
Tvrdé kolize (z angl. *hard clashes*) zahrnují ryze geometrické konflikty prvků, které mají za následek vzájemné protínání bez jakéhokoliv upřesnění spoje či průchodu. Mezi obecnými typy kolizí jsou tyto pravděpodobně nejlépe rozpoznatelné. Nejen z důvodu logického uspořádání prvků, ale i z jejich nulové dynamičnosti – neuvažuje se u nich s pohybem a lze tak počítat s jejich stálým umístěním v konstrukci a spolupůsobením s ostatními prvky. Jako konkrétní příklady některých tvrdých kolizí můžeme uvést tyto:

- Konflikt mezi potrubím stěnou bez definovaného prostupu (obrázek 3)
- Schodiště prochází vodorovnou konstrukcí, která nebyla náležitě oříznuta (obrázek 4)

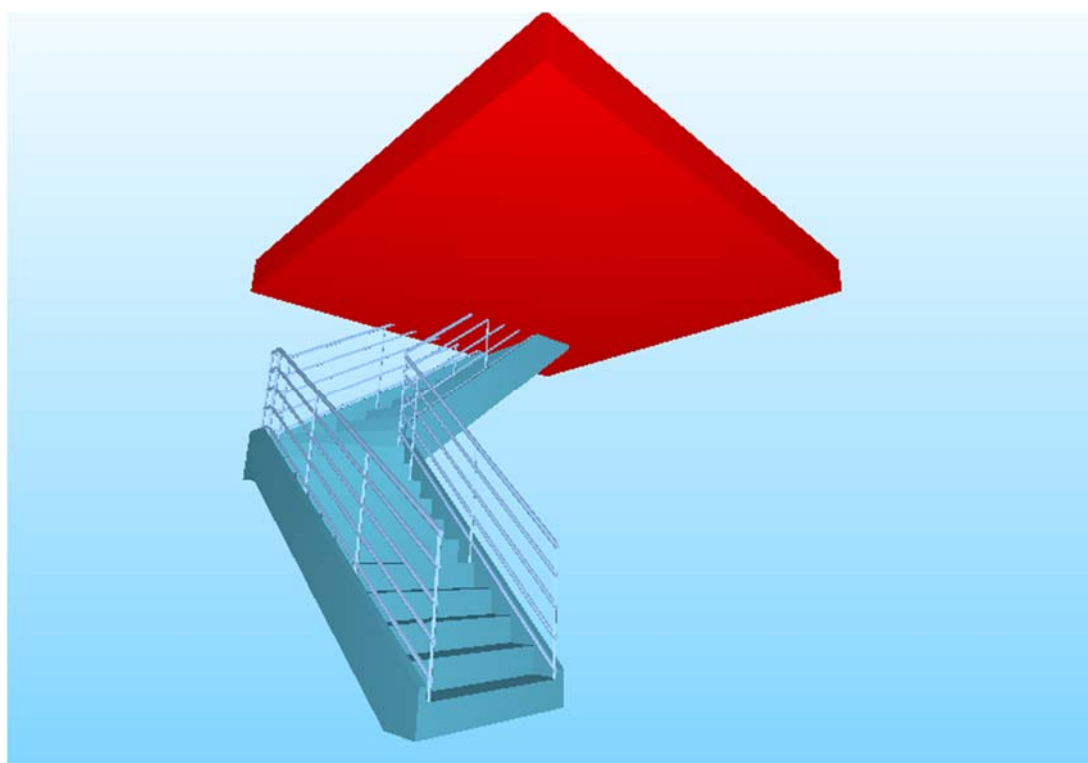
Mezi časté chyby modelování, vedoucí ke vzniku tvrdých kolizí, patří například špatné připojení příčky na nosnou zeď, kdy příčka prochází nosnou zdí nebo usazení klozetu ve vzdálenosti nižší než je poloměr dveřního křídla, otvíravého směrem k toaletě. Do konfliktu se mohou snadno dostat předměty statického modelu, jako jsou například nosníky s prvky modelu TZB (potrubí či vedení vzduchotechniky).

Tato kategorie se řadí mezi závažné a je třeba k ní takto přistupovat. Každý výskyt takové kolize je tudíž třeba neprodleně řešit, neboť představují výraznou potenciální komplikaci projektu. Je důrazně doporučováno, soustředit se na její co nejrychlejší odstranění. Systém autorizace projektantů a architektů, v současné době, napomáhá řešení kolizí již při samotném návrhu konstrukcí. Drtivá většina konfliktů je tak řešena včas a nedostává se do dalších fází kontroly projektu. Nicméně se ale mohou objevit výjimky, které mohou být zapříčiněny nedostatečnou praxí či profesionalitou projektanta, špatnou koordinací mezi projektanty profesí TZB, statikem, architektem a projektantem nebo pouhou nepozorností, případně nedůsledností některého z výše jmenovaných.

Mezi tvrdé kolize patří i speciální kategorie konfliktů, kterými jsou duplikované konstrukce. Během tvorby 3D modelu může nastat případ, kdy je omylem přidán stavební prvek na místo, kde se už jeden stejný nachází. Současné nástroje na tvorbu modelů (Revit, ArchiCAD,...) dokáží, hned po vložení prvku, upozornit na výskyt duplikátu. Ne vždy je ale toto opatření dostačující a zdvojené konstrukce zůstávají v modelech a pokračují do dalších fází. Takové duplikace nejsou však ničím jiným, než tvrdými kolizemi dvou a více shodných prvků na naprosto stejném místě a tudíž jsou v rámci vyhledávání kolizí zpravidla spolehlivě určovány. Jedná se o typ kolizí, který se objevil až díky objektově orientovanému modelování a zpravidla není obtížné mu předejít. V případě nepozornosti však může způsobit nepřesnosti například ve výkazech výměr.



**Obrázek 3: Ukázka tvrdé kolize mezi potrubím a stěnou**  
Zdroj: vlastní tvorba



**Obrázek 4: Ukázka tvrdé kolize mezi schodištěm a vodorovnou konstrukcí**  
Zdroj: vlastní tvorba



### 1.3.1.2 Měkké kolize

Měkké kolize (z angl. *soft clashes*) nejsou, na rozdíl od tvrdých kolizí, vázány pouze geometrickými vlastnostmi. V tomto případě jsou již do kontrolního procesu zahrnovány i prostorové nároky jednotlivých prvků, postupy napojení nebo uložení konstrukcí nebo požadavky na prostor pro manipulaci. Pro definici těchto nároků se často používá pojem *buffer*<sup>1</sup>. Ten je u měkkých kolizí statický, na rozdíl od kolizí animovaných předmětů, viz kapitola 1.3.1.4.

„Měkká kolize odkazuje na zásah jednoho systému do bufferu, nastaveného kolem systému druhého.“<sup>2</sup> [8, s. 187]

Mezi měkké kolize je možné zařadit tyto případy:

- Nedostatečné uložení nosníků, překladů (obrázek 5)
- Umístění konstrukce (sloup, stěna) do prostoru před oknem, dveřmi nebo průchodem (obrázek 6)
- Nedostatečná vzdálenost před zařizovacími předměty (obrázek 7)
- Nedostatečný rozměr podesty či šířky ramene schodiště (obrázek 8)

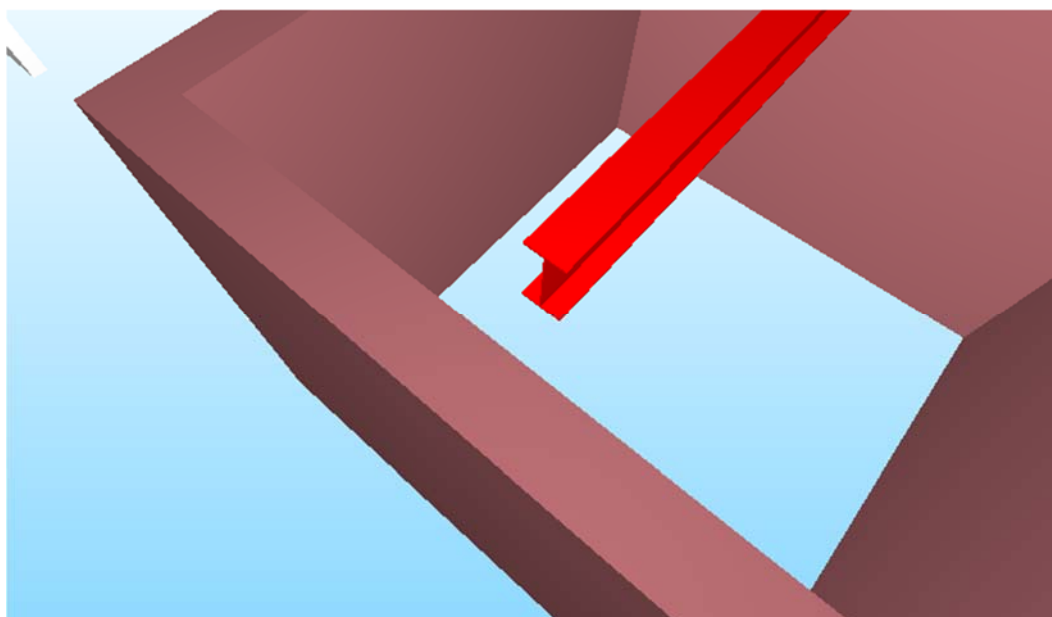
Dalšími chybami návrhu, které spadají do kategorie měkkých kolizí, mohou být například nedodržené předepsané rozměry schodišť nebo manipulační vzdálenosti kolem zařízení MAR, požárních hydrantů, apod. Do této kategorie patří i kontrola minimální světlé výšky místnosti nebo podchodné a průchodné výšky na schodišti.

Velká část měkkých kolizí je řešena z důvodu správného užívání prostor. Algoritmus, který se používá na vyhledávání takových nedostatků, vychází z určitých pravidel, jenž definují správnost návrhu z hlediska bezpečného pohybu a samotné možnosti užívání objektu. Oproti tvrdým kolizím se sice nemusí nutně jednat o zásadní chybu v projektové dokumentaci, kvůli které by nebylo možné stavbu dokončit nebo používat. V některých případech je možné měkké kolize tolerovat i za cenu následného omezení v užívání. Tato možnost by měla být zvolena pouze v případech, kdy se jedná o relativně nepodstatný nedostatek, u kterého finanční náklady na odstranění nebo případný časový posun výstavby významně převažují omezení, které nastane jeho zachováním. To je opět případ, kdy je nutné podpořit automatizovanou kontrolu kolizí lidským faktorem a přispět tak k co nejlepší stavební praxi.

---

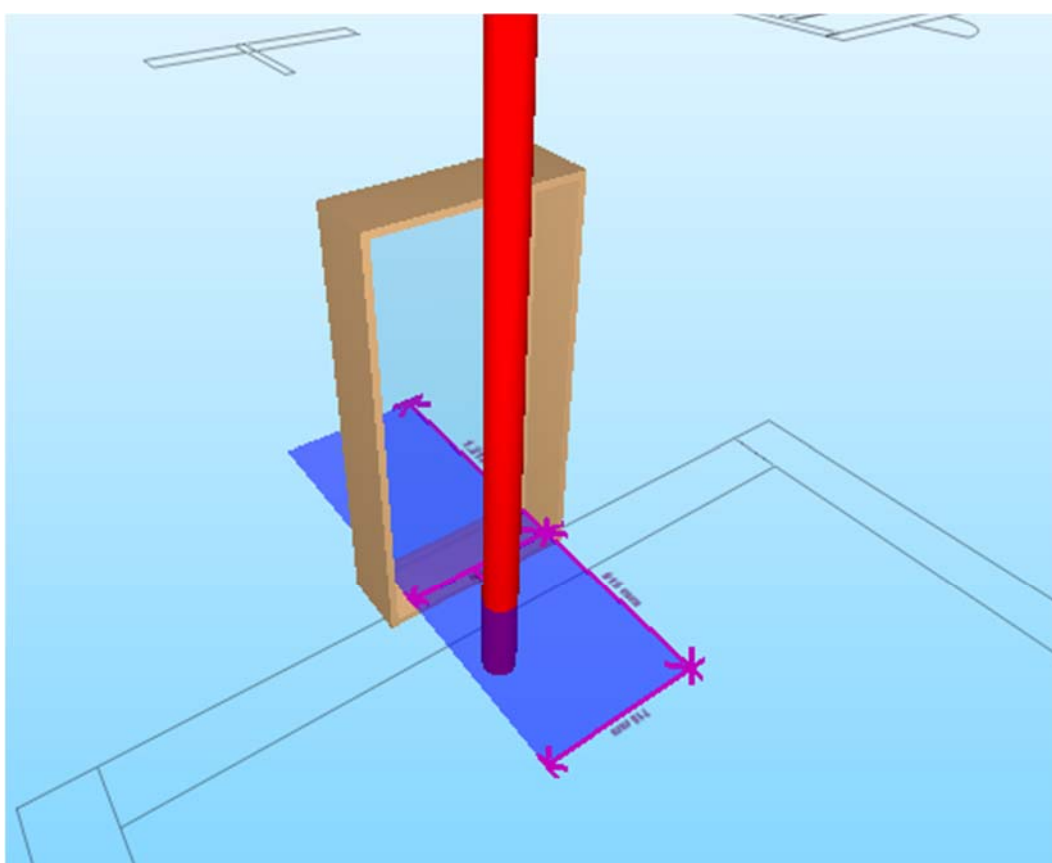
<sup>1</sup> Pojem *buffer* je používán i v programech GIS (*Geographical Information System*), ve kterých definuje zónu kolem určitého objektu na mapě.

<sup>2</sup> Volný překlad.



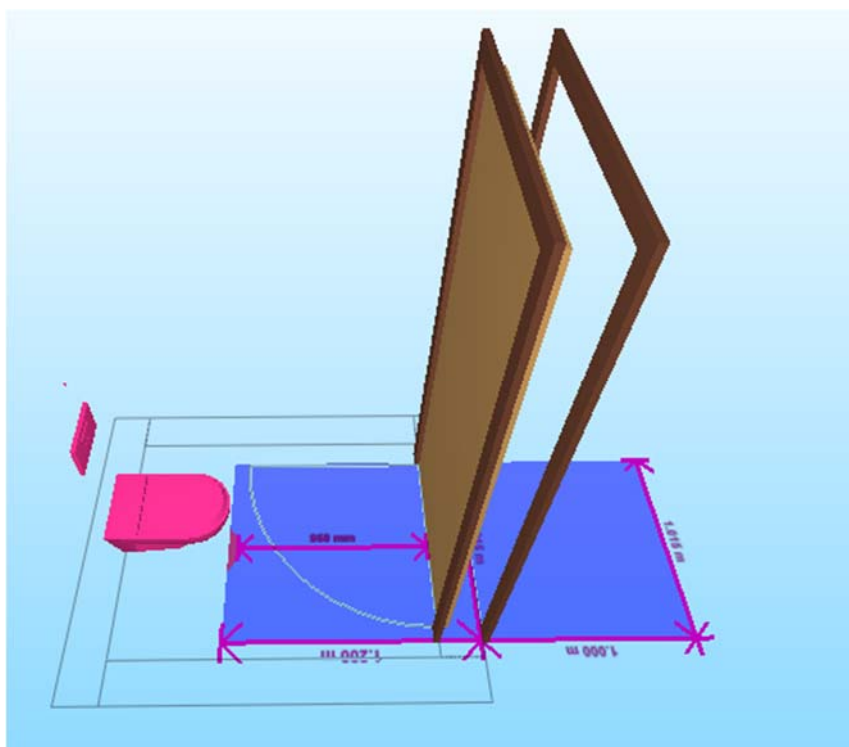
**Obrázek 5: Ukázka měkké kolize nedostatečného uložení nosníku**

Zdroj: vlastní tvorba

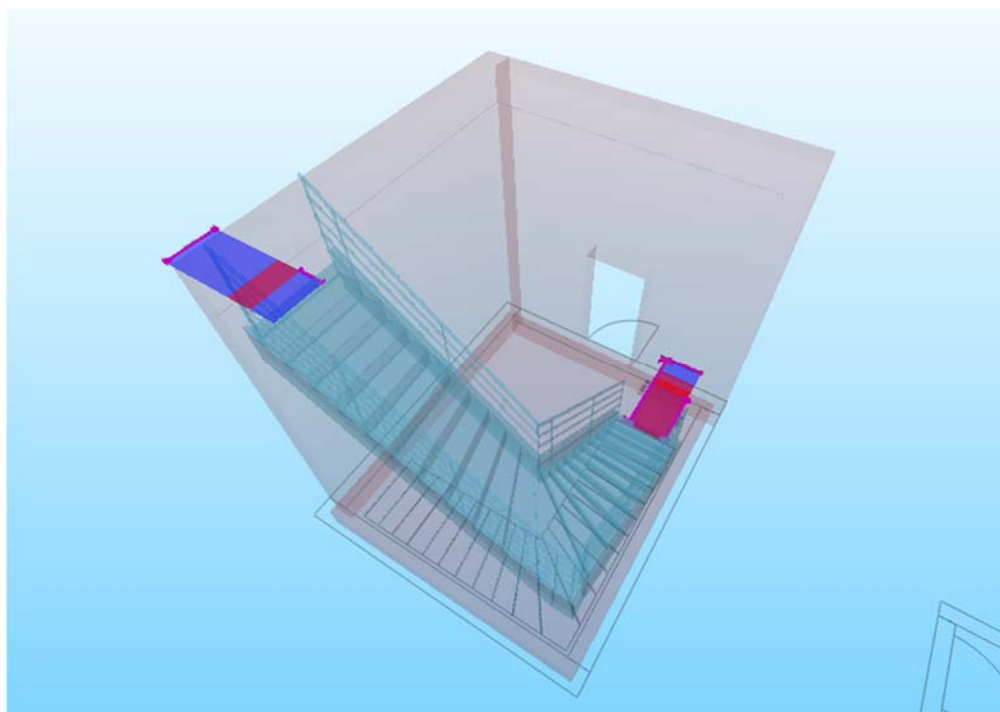


**Obrázek 6: Ukázka měkké kolize sloupku umístěného před prvek okna**

Zdroj: vlastní tvorba



Obrázek 7: Ukázka měkké kolize nedostatečného prostoru mezi klozetem a dveřmi  
Zdroj: vlastní tvorba



Obrázek 8: Ukázka měkké kolize v nedostatečných rozměrech před nástupním a výstupním stupněm schodiště  
Zdroj: vlastní tvorba



### 1.3.1.3 4D kolize

Existuje mnoho případů nesprávného dodržení časového plánu či koordinace prací, které mohou ovlivnit probíhající stavbu. Těmto případům se věnují 4D kolize. Jde o typ konfliktů, které nastávají během stavební výroby a které zpravidla nejsou přímo spojeny s návrhem konstrukce. Mezi 4D kolize můžeme řadit například:

- Nedostatečná vzdálenost zvedacího zařízení od hranice stavby
- Hromadění uskladněného materiálu
- Špatně projektované zařízení staveniště z hlediska manipulačního prostoru pro nákladní automobily a těžkou mechanizaci
- Nástup pracovní čety na pracoviště, kde není připraven materiál
- Nástup pracovní čety na pracoviště, kde nedokončila svou práci předchozí četa
- Konflikt zvedacího zařízení s nově vystaveným lešením

Příčinou některých 4D kolizí bývá převážně nedůslednost při provádění stavby. Nástroje, které lze využít pro kontrolu možných budoucích problémů mají funkci pouze preventivní. Nemůžeme se na ně plně spolehnout, je třeba zajistit plnění stanovených doporučení. Softwarová kontrola však může přinášet určité benefity při projektování zařízení staveniště. V případě, že projektant umístí jeřáb v nedostatečné vzdálenosti od objektu, systém konflikt rozpozná a upozorní na něj. Tím se dá docílit optimalizace přidruženého projektování. Problémem však mohou být relativně individuální podmínky výstavby a jejich stavenišť. Je pak třeba do pravidel, které program využívá pro detekci 4D kolizí, zavést požadované hodnoty i s ohledem na budoucí výstavbu.

### 1.3.1.4 Kolize animovaných objektů

Do této kategorie mohou spadat kolize, které jsou určovány z hlediska dynamičnosti komponent. To zahrnuje animace předmětů reprezentujících pohyb osob a věcí (schodiště, průchozí koridory, atd.) nebo komponenty, se kterými je zapotřebí během užívání manipulovat. Prakticky tento typ kolizí vychází z měkkých kolizí s tím rozdílem, že jejich buffer již není statický, ale stává se dynamickým.

Kolizi animovaných předmětů si lze představit například jako konflikt mezi stavbou a autojeřábem, který byl přesunut z původní nekolizní polohy, nebo vymodelovanou postavou, která prochází koridory a kontroluje tak jejich předepsané rozměry. Příslušný algoritmus pro definování pohyblivého bufferu tak bude jiný a zpravidla složitější než u měkkých kolizí a s nimi spojených statických bufferů.





### 1.3.2 Dopady kolizí

Jak již bylo naznačeno v předchozích kapitolách, představují kolize potenciální hrozbu. Výhodou práce s BIM modelem je funkce virtuální simulace výstavby a instalace jednotlivých komponent. Takový postup nabízí uživateli zásadní možnost včasné přípravy na případná rizika spojené s chybami v návrhu.

Podle Kymmella je detekce kolizí nejefektivnějším nástrojem pro virtuální koordinaci prací a měla by být integrovanou součástí práce s BIM modelem [9]. Podobný názor sdílí i Reddy, který detekci kolizí označuje za leč základní, ale extrémně cenný způsob využití BIM technologií. I když autor nevyklučuje možnost operativního řešení kolizí in-situ, je ale stále zdůrazňován negativní dopad na finanční a termínovou stránku projektu. [8, s. 187]

Je obecně považováno za nejlepší, věnovat se kolizím a podobným nedostatkům v návrhu co nejdříve. Pokud není kolize řešena již v počátečních fázích, fyzicky se objeví na staveništi, což bude mít za následek náklady na změnu projektu a jeho zdržení [8, s. 185]. Ignorování podstatných kolizí vede k téměř jistému navýšení stavebních nákladů na opravy a přepracování. [10]

Existuje mnoho případů, kdy stavební firmy vykazují značné úspory díky použití BIM metodiky při vyhledávání chyb v návrhu. Dle společnosti *BAM* bylo na projektu výstavby Leeds Arény ve Velké Británii ušetřeno 350 000 liber právě použitím detekce kolizí, díky které byly konflikty vyhledány včas a negenerovaly tak další náklady. [11] Podobnou zprávu uvedla firma *Holder Construction*, která prohlásila, že na jednom ze svých menších projektů ušetřila díky detekci kolizí 135 000 amerických dolarů s investicí do BIM prostředků 4 000 dolarů. [12] Společnost *Tesseract-Design* uvedla vyčíslení nákladů na předělání jednoho neřešeného konfliktu, a to ve výši 750 amerických dolarů. [13] Autoři dále uvádí úspory díky detekci kolizí na projektu menší multifunkční budovy ve výši 9 000 dolarů, a s náklady 2 945 dolarů na koordinaci v programu *Navisworks* hovoří o ROI (*Return On Investment*) ve výši 306%. Na jiném, větším projektu činily celkové náklady 43 402 dolarů, úspory 127 850 dolarů a ROI 295%. Autoři tak hovoří o 300% návratnosti investic do BIM systémů pro detekci kolizí. [13]

V současné době se tak začíná metodika BIM využívat i z hlediska oboru BOZP. Opomenutí zdánlivě nepodstatných konfliktů však může vést i k nebezpečným situacím, ve kterých hrozí pracovníkům úraz či dokonce smrt. Ve Spojených státech amerických vyvinuli inženýři systém GPS vysílačů v bezpečnostních vestách pracovníků, které po napojení na informační model stavby vydávají zvukový a vizuální signál v případě, kdy systém zaznamená vstoupení do předem zakázané zóny. [14] Tímto způsobem lze dohlížet na bezpečnost pracovníků ve zcela novém smyslu.



### 1.3.3 Řešení kolizí

V praxi se mohou kolize objevit v různých etapách projektu. Některé se dají řešit hned v počáteční fázi, což je ekonomicky nejvhodnější, jak již bylo zmíněno v úvodní kapitole (kapitola 1.1). Zejména během rekonstrukcí se však může objevit dříve nedetekovaný konflikt až teprve při samotném provádění stavby. Tyto problémy je potom třeba řešit operativně. To je možné mnohdy pouze na místě, případně s podporou přenosné výpočetní techniky. Ideou softwarových vývojářů je zakomponování virtuální prohlídky BIM modelu pomocí tabletu přímo in-situ. Odpovědný pracovník pak může kontrolovat právě probíhající práce a hned řešit případné kolize. Taková kontrola skýtá potenciál ve značném zrychlení odstraňování chyb v návrhu, které byly při předchozích revizích opomenuty.

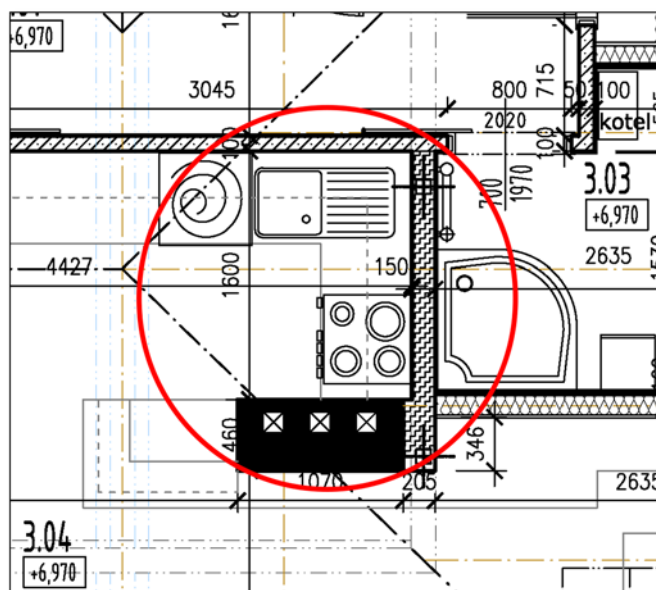
Původní formou kontroly kolizí (převážně tvrdých a měkkých) bylo překládání výkresů přes sebe, podsvícenými na speciálním stole – například výkres vzduchotechniky s výkresem architektonického řešení. Oproti kontrole pomocí počítače se však jedná o chybovou a zdoluhavou metodu [15, s. 273]. Kolize se mnohdy řeší i během samotného průběhu stavby. Je však velmi důležité skloubit a zkoordinovat činnosti všech profesí.

Projektant či architekt musí brát v potaz veškeré konstrukce, které se na stavbě vyskytují před zahájením a během stavebních prací. Z hlediska ideální funkčnosti návrhu je pak zapotřebí se zaměřovat na takové prvky, které v objektu přibydou až ve fázi užívání. Příkladem může být samotný návrh dispozice obytné místnosti – architekt musí přemýšlet, zda se do vyhrazeného prostoru vejdou objekty jako jídelní stůl, židle apod. a navíc bude kolem nich dostatek manipulačního prostoru.

Při běžném způsobu řešení kolize, která vznikla na projektu rekonstrukce, je zpravidla dobré postupovat určitým způsobem. Je nutné si přímo na stavbě zaměřit dotčené konstrukce. Dále je nezbytná jistá znalost technologií a provádění staveb. Míra záleží na závažnosti kolize. Důležitou součástí postupu je dostatečná představivost konstrukcí, které se budou na stavbě postupně objevovat. K tomu je třeba dokonale znát samotný projekt. Pokud je komplikovanost kolize přílišná a nedá se řešit na místě, nabízí se zastavit související práce, pořídit fotografie místa konfliktu a přepracovat návrh pomocí programů na projektování.

Jako příklad podobné kolize a postupu řešení lze uvést křížení vedení odtahu odsavače par a dřevěných konstrukčních prvků krovu a střešního výlezu. K této kolizi došlo na rekonstrukci rodinného domu a byla řešena následujícím způsobem:

Počátečním umístěním odsavače par nad indukční desku byla určena trasa vedení odvětrávacího potrubí kolmo nahoru, do střechy se zakončením větrací střešní taškou. V půdorysném pohledu (obrázek 9) však nebyl zakreslen výlez na střechu, který se nacházel v poli mezi krokveji, přímo nad odsavačem par.



Obrázek 9: Detail půdorysu podkrovního bytu se zaměřením na místo kolize vedení vzduchotechniky

Zdroj: vlastní tvorba s použitím výkresu [16]

Na stavbě bylo zdokumentováno (obrázek 10) a (obrázek 11) umístění kolizních prvků a odměřeny vzdálenosti mezi koncem uvažovaných horních skříňek kuchyně a spodních hran trámů mezi vaznicí a kleštinou.



Obrázek 10: Fotografie prostoru kolize vedení vzduchotechniky, pohled z kuchyně

Zdroj: vlastní tvorba



**Obrázek 11: Fotografie prostoru kolize vedení vzduchotechniky, pohled z koupelny**

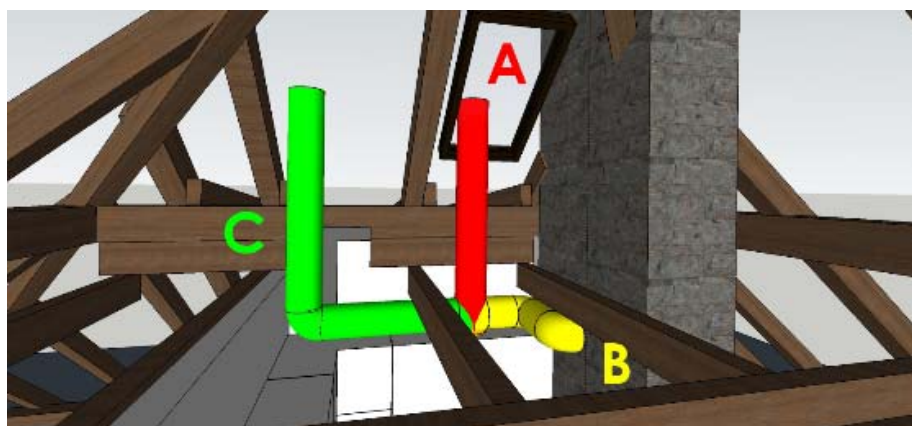
Zdroj: vlastní tvorba

Jako podklad pro rozhodování byl vytvořen model (obrázek 12) s možnými variantami vedení vzduchotechniky:

A = Původní varianta. Nemohlo být přijato kvůli kolizi vedení a střešního výlezu.

B = Návrh odtahu do komína. Kvůli nutnosti připojení do středu komínového tělesa a zakrytí potrubí sádrokartonem v prostoru kuchyně, by došlo ke vzniku nevzhledného a tudíž architektonicky nepřijatelného objemu.

C = Návrh odklonění vedení do pole mezi sousedními krokvi. Vzhledem k bezproblémovému zakrytí konstrukce a snadnému odvodu potrubí vedle střešního výlezu byla vybrána tato varianta.



**Obrázek 12: Názorné zobrazení variant vedení vzduchotechniky**

Zdroj: vlastní tvorba



## 2 Softwarové nástroje pro detekci kolizí v BIM modelu

Jedním z cílů této práce je průzkum počítačových programů, umožňující práci s modelem, která se zaměřuje především na detekci kolizí mezi stavebními prvky. Část nástrojů však kombinuje různé funkce, což může ve výsledku vést k potlačení primárního využití programu jako vyhledávače konfliktů. Kvůli tomuto faktu byla stanovena určitá kritéria týkající se především typu softwarové aplikace a kompatibility datových formátů. Jejich smyslem bylo alespoň částečně vymezit okruh nástrojů, který byl posléze podroben důkladnějšímu zkoumání. Tato kritéria byla popsána a odůvodněna v následující kapitole.

### 2.1 Kritéria výběru softwaru

#### 2.1.1 Stand-alone software

V současnosti se začíná značně rozrůstat rodina softwarových nástrojů, které se dají použít k vyhledávání kolizí. Z počátku byla tato funkce obsažena jen v programech, specializujících se na projektování a tvorbu 3D modelů staveb. Jednalo se spíše o podpůrnou činnost a možnosti programování detekce kolizí byly značně omezeny.

S rozvojem informačního modelování se však začala tato oblast práce s modelem dostávat do popředí. To neodvratně vedlo k vytváření sofistikovanějších metod a přístupů k řešení kolizí. Neznamenalo to pouze zlepšování integrovaných funkcí programů k projektování. Přímým důsledkem tak bylo vytvoření nové kategorie počítačových aplikací. Ty mají zpravidla výrazně menší rozsah a užší specializaci. Pracují se soubory, které jsou vytvořeny projektanty různých profesí v odlišných programech. Na rozdíl od integrovaných funkcí se jedná o již, zcela nezávislé tzv. *stand-alone* programy.

Výhodou takových programů je mimo jiné i to, že svým konkrétním zaměřením zpravidla předčí integrované funkce komplexnějších nástrojů. V praxi to tak může znamenat, že detekce kolizí ve specializovaném programu (např. *Solibri*) může vykazovat mnohem lepší výsledky než sekundární funkce v modelovacích programech (*Revit*, *ArchiCAD*, apod.). Tyto funkce jsou zakomponovány do modelovacího *work-flow* a byť byly zmíněné modelovací programy ochuzeny o vestavěné aplikace, byly doplněny o systém kontroly přímo v průběhu tvorby 3D modelu. V případě hrozícího konfliktu je tak uživatel upozorněn a pro pokračování v modelování je třeba hlášku odsouhlasit.

Komplikací pro *stand-alone* typy programů může být i v počtu potřebných licencí. Samostatnou aplikací společně s vhodně upraveným přenosem dat je možné docílit určité nezávislosti na dodavateli projekčních programů. Na druhou stranu mohou



vznikat náklady při navyšování licencí pro jednotlivé účastníky projektu. Tento problém by mohlo řešit používání BCF formátu, o kterém je pojednáno v jedné z následujících kapitol (kapitola 2.4.3).

V rámci výběru dostupných softwarových nástrojů v této práci, byla tak programová nezávislost stanovena prvním kritériem. Smyslem je určit takový software, který může být použit bez ohledu na to, jaký program či dodavatel softwaru je právě používán k vytváření modelů a dalších BIM operací.

### 2.1.2 IFC

Postupem času se ukázalo, že rozvoj aplikací vede i k nepřiměřeně velkému množství používaných formátů. To vedlo k potížím se spojováním souborů od jednotlivých projektantů, kteří každý používali rozdílné programy a jejich specifické formáty. Podle Eastmana a kol. existují dvě varianty, jak se s tímto problémem vyrovnat [15, s. 18]:

- Používat jednoho dodavatele softwaru v rámci celé skupiny
- Používat rozdílné programy, které ale mezi sebou dokáží přenášet informace pomocí standardizovaných datových formátů.

Dle autorů má první varianta výhodu ve snazší koordinaci, převodu modelu z jednoho programu do druhého a jejich možného propojování. Jako nevýhodu však uvádí nutnost použití všech programů od stejného dodavatele. Druhý přístup shledává Eastman a kol. flexibilnějším. Stinnou stránkou je ale hrozba ztráty informací během převodu modelu do standardizovaného formátu. Tomu je však možné do jisté míry zamezit jasnou specifikací standardů pro výstupy v průběhu celého životního cyklu stavby. Autoři tak zmiňují využití uzavřených či *open-source* standardy, mezi které patří *Industry Foundation Classes* (IFC).

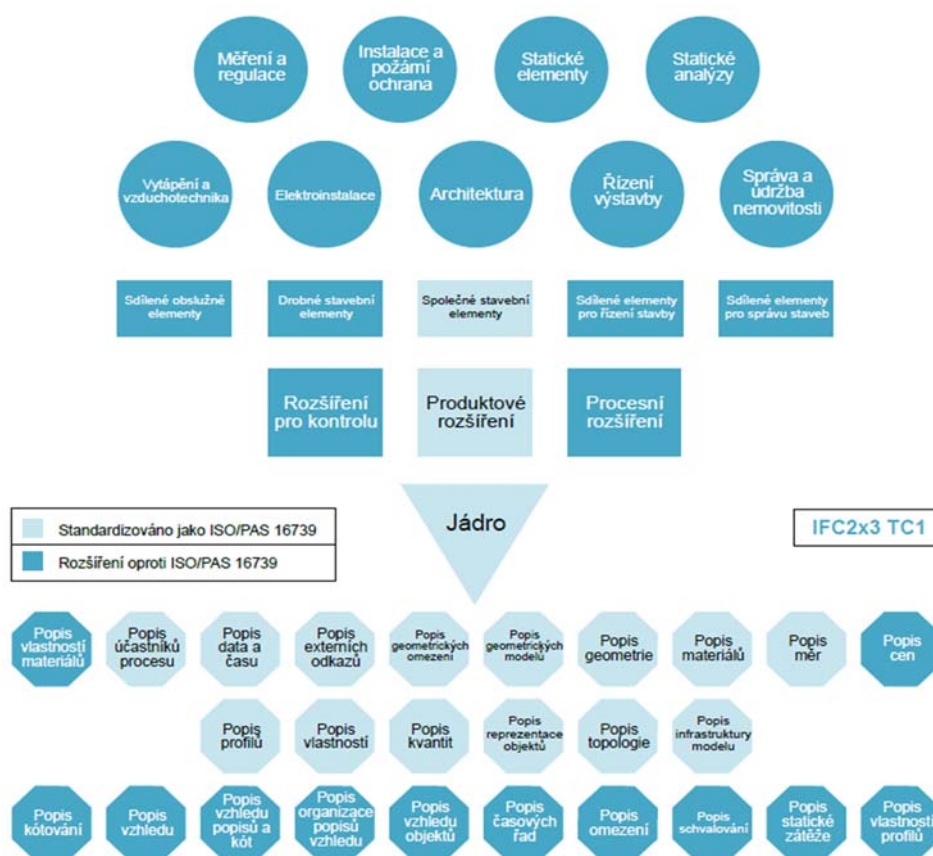
IFC formát byl vytvořen společností *buildingSMART International* (dříve *International Association for Interoperability*) v roce 2006 a registrován ve verzi IFC 2x3 jako norma ISO 16739, jež byla následně v roce 2013 aktualizována ve verzi IFC4. [1] Rozsah a struktura IFC je znázorněna v diagramu (obrázek 13). Důležitost IFC vidí autoři v jeho otevřenosti. Účelem je podpora vývojářů při tvorbě vlastních aplikací, operujících s tímto formátem. Díky tomu je také snazší manipulace a práce se soubory jak na straně autora návrhu, tak na straně uživatele objektu.

Předností formátu IFC je podle Černého a kol. také schopnost zahrnout prakticky všechny informace o budově a poskytovat je i budoucím nástrojům, které je, oproti těm současným, budou umět využívat. Jistou nevýhodou použití IFC však může být nadměrná velikost exportovaných souborů. Tento problém se dá do určité míry eliminovat zavedením formátu *ifczip*. Tento upravený datový formát umožňuje zkomprimovat

soubory IFC tak, že výsledná verze dosahuje zhruba 40-20% velikosti původního obsahu [1]. Není však pravidlem, že by program, běžně pracující s IFC, dokázal číst a manipulovat s daty ve formátu *ifczip*. Zde je prostor pro potenciální zmenšení datových nároků a celkovou optimalizaci práce se standardizovanými soubory.

Otevřený formát IFC skýtá i možnosti zkvalitnění a automatizace projektování. Během své práce mohou projektanti využít IFC modelů stavebních prvků, které jsou vyvíjeny přímo jejich výrobci. Tímto postupem vznikají tzv. rodiny a knihovny rodin. Pro celkové sjednocení BIM postupů je však třeba legislativního uspořádání a definice používaných prvků. V současné době se knihovny produktů, obsahující jejich vlastnosti a specifikace, zabývá norma ISO 16757 [1].

Vzhledem k zásadní důležitosti formátu IFC při práci s modely z různých odvětví stavebního průmyslu byla příslušná kompatibilita dalším kritériem při výběru programů pro detekci kolizí, zkoumaných v této práci. Zásadní se jeví převážně pro svou standardizaci a kompatibilitu. Proto se dá očekávat jeho šíření a běžné používání i v budoucnosti.



Obrázek 13: Schématické znázornění rozsahu IFC vzhledem k ISO 16739

Zdroj: Černý a kol. [1]



## 2.2 Přehled vybraných programů

Samotný průzkum trhu se softwarovými nástroji byl zaměřen primárně na takové programy, u kterých jejich výrobci uváděli vlastnosti, které splňovaly výše zmíněná kritéria. Na základě rešerše literatury a konzultací s praxí bylo záměrem zahrnout do výběru nejen obecně používané nástroje renomovaných vývojářských společností, ale prozkoumat i dostupnost a možnosti méně známých programů. Jako výstup průzkumu byla sestavena tabulka s okomentovanými vlastnostmi vybraných programů. Tyto vlastnosti byly určeny za účelem vytvoření obecného přehledu možností, které programy nabízí v oblasti, jíž se zabývá tato práce. Jde tedy především o kompatibilitu datových formátů, uživatelské rozhraní při manipulaci s modelem, možnosti detekce kolizí, jejich revidování, delegování odpovědných osob, poskytování reportů a v neposlední řadě i cen

## 2.3 Vlastnosti vybraných programů

Jednou z komentovaných vlastností vybraných programů je typ možného zobrazení modelu. Tato funkce může být považována za prakticky nepodstatnou pro svou jednoduchost. Nabízí však různé možnosti zobrazení mnohdy velmi komplikovaných částí modelu. Dobrá orientace při řešení složitých vazeb mezi jednotlivými systémy je kritická. Při nedostatečném zobrazení může být však velmi složitá a následkem může být i naprostá nepoužitelnost programu. Základní funkcí je volba průhlednosti konstrukcí, čímž může uživatel zobrazit polohu jinak skrytých prvků. Dalším nástrojem je pohledové řezání konstrukcí. Ořezovými rovinami lze opět zobrazit prvky a názorně ukázat jejich návaznosti na ostatní systémy. Z hlediska zobrazení kolizních prvků jsou velmi vhodné i funkce izolace a skrývání vybraných komponent.

V tabulce je dále zmíněna schopnost programu zobrazovat a pracovat s animovanými předměty. Tuto funkci lze z hlediska detekce kolizí využít nejvíce při simulování a kontrole 4D kolizí. Například lze tak animovat pohyb zdvihacího či dopravního prostředku a kontrolovat jeho případné konflikty se statickou částí modelu, která je zpravidla tvořena virtuální stavbou. S tím souvisí i simulace výstavby, která může pomoci při kontrole harmonogramu nebo koordinaci prací.






Pojmem *BIM data mining* je v této práci označována sekundární funkce nástrojů pro export informací z modelu. Může se jednat o různé počty, tabulky či výkazy, které lze využít dále například v controllingu nebo při tvorbě rozpočtu stavby.





Tabulka 1: Přehled vybraných programů pro práci s kolizemi v BIM modelech

Zdroj: vlastní tvorba

	<b>Navisworks Manage</b> 	<b>Tekla BIMSight</b> 	<b>Solibri Model Checker</b> 	<b>BIMReview</b> 	<b>Bentley Navigator Connect</b> 
Vývojář	Autodesk, Inc.	Tekla, Trimble	Solibri, Inc.	Strumis LTD.	Bentley Systems
Podpora IFC	ano	ano	ano	ano	převodem
Další import formáty	většina projekčních formátů	DWG, DGN	DWG, IFCzip, zip	stp, igs, ifczip, i.dgn, xml, gtcx, asproj, bsw	i.dgn
Formáty exportu	NWD, NWF, DWF, FBX, KML	tbp	SMC	BSWX, IFC	i.dgn - cloud
Zobrazení modelu	průhledné, izolace, skrývání, řezy	průhledné, vrstvy, izolace	průhledné, selection basket - izolace, řezy	izolace, skrývání, řezy	izolace, skrývání, řezy
HMG, animace	ano	ne	ne	ano	ano
BIM data mining	ano	ne	ano	ne	ne
Programování kolizí	pokročilé	pouze základní	pokročilé	pouze základní	omezené
Zadávání pravidel	standardní	omezené	přehledné	komplikované	standardní
Označování kolizí	nová, aktivní, zaznamenaná, odsouhlasená, vyřešená	nová, přetrvávající, přiřazená, kritická, vyřešená, nepodstatná	kritická, středně důležitá, méně důležitá + přijatá, odmítnutá, nedefinovatelná, neovlivnitelná	nová, zaznamenaná, nepodstatná, přiřazená, vyřešená, kritická	dle priority - naléhavá, vysoká, střední, nízká
Zvýraznění kolizí	ano, 3D poznámky	ano, 3D poznámky	ano, 3D poznámky	3D poznámky	ano, 3D poznámky
Delegování odpovědných pracovníků	přiřazení řešitele, komentáře	komentáře	přiřazení řešitele, komentáře	komentáře	komentáře
Typ exportu a komunikace	reporty, animace	pouze detaily kolizí - obraz	video - prezentace, tabulky, reporty	pouze detaily kolizí - obraz	komentáře, detaily kolizí
Formát reportu	XML, HTML, slideshow	slideshow	export - BCF, RTF, PDF, Excel	obrazy	obrazy
Orientační cena <sup>3</sup>	<u>2255 EUR ročně</u> [17]	<u>zdarma</u> [18]	<u>1500 EUR / 90 dní</u> [19]	2000 EUR [20] <sup>4</sup>	11 500 Kč + 2 300 Kč/rok [21] <sup>5</sup>

<sup>3</sup> Ceny aktuální ke dni 24. 11. 2016<sup>4</sup> Cena aktuální ke dni 5. 12. 2016<sup>5</sup> Orientační cena dle CAD Lexikon



Nejdůležitější vlastností, která byla v tabulce (tabulka 1) rozdělena do dvou částí je práce s kolizemi. V první řadě je třeba sledovat, do jaké míry a podrobnosti lze pomocí softwaru nastavovat samotná pravidla pro detekci kolizí. Z důvodu značně odlišných přístupů jednotlivých programů byla tato vlastnost okomentována pouze slovně. Dalším faktorem je uživatelská vstřícnost aplikace kolizních pravidel. Software sice může umět tvořit sofistikovaná pravidla. To ale stále není zárukou jejich následné logické aplikace. V ideálním případě by měl být uživatel schopný detailně nastavit kolizní pravidla a zároveň je přehledně aplikovat na model. Taková aplikace by se měla projevit v názornosti a transparentnosti výsledků, kdy uživatel přesně dokáže určit, které pravidlo generovalo konkrétní kolizi a ideálně kterých prvků se týká.

Důležitou součástí pro práci s kolizemi je pak jejich identifikace, zařazení a označení. V případě, že člověk dokáže pomocí programu nalézt, ale nedokáže jí přiřadit důležitost a upozornit na ni, většina přínosu detekce přichází vniveč. Uvedené softwarové nástroje používají různé kategorie pro určení stupně důležitosti kolize. Obecně se však shodují na podobné struktuře dle závažnosti konfliktu a úrovně řešení.

Dále je potřeba kolizi vizuálně zvýraznit. To je podstatné pro následné delegování odpovědných pracovníků. Zde programy nabízí přidávání komentářů ke konfliktům nebo přímé určování jejich řešitelů. Další uživatelé pak mohou běžně přidávat své doplňující komentáře a odpovědi, čímž se obecně zpřehlední práce při odstraňování kolizí.

Pro práci s kolizemi v rámci týmu je podstatnou součástí tvorba reportů. Softwarové nástroje shodně podporují tvorbu obrazové dokumentace s výše uvedenými komentáři. V některých případech je ale tato funkce doplněna o tvorbu animací. Ty mohou pomoci při sestavování prezentací a tím ještě více podpořit orientaci mezi komponenty modelu. Jednou z velice užitečných forem reportu je pak export do tabulkového procesoru (např. *MS Excel*), kde lze s informacemi dále pracovat. U některých programů lze také využít exportu informací ve formátu BCF (*BIM Collaboration Format*).

Nástroje uvedené v tabulce můžeme doplnit o další programy, které dle jejich vývojářů splňovaly požadovaná kritéria, ale nebylo z různých důvodů možné je vyzkoušet a ověřit tak uvedené informace. Konkrétně byla vyvinuta snaha o získání přístupu k těmto programům:

- *Synchro Pro*
- *DDS CAD Viewer*
- *IFC Viewer*

Ty mohou tvořit potenciální konkurenci výše uvedeným nástrojům. Zde se tak nabízí prostor k dalšímu průzkumu a podrobnějšímu doplnění rešerše.



## 2.4 Popis vlastností porovnávaných programů

V této kapitole jsou popsány vlastnosti a nastíněn způsob práce nástrojů na detekci BIM kolizí, které byly uvedeny v přehledové tabulce (tabulka 1). Postup ve zkoumání *work-flow* spočíval v importu jednoduchého projektu, předem zpracovaného autorem v BIM modelovacím nástroji *Autodesk Revit Architecture*. Tím byla zajištěna objektivní porovnatelnost vlastností zkoumaných nástrojů. Samotné popisy programů jsou doplněny poznámkami s komentáři autora.

Z důvodu problematického převodu *Revit Architecture* modelu do *i.dgn* formátu, potřebného pro práci v programech společnosti *Bentley Systems*, a to i přes použití oficiálního postupu výrobce, zahrnujícího využití zásuvného modulu *Bentley* pro *Revit Architecture*. *Bentley Navigator Connect* tak nemohl být řádně zkoumán a nebyl tudíž zahrnut do kapitol podrobného popisu. Je možné na tuto práci navázat detailním náhledem na *work-flow* v *Bentley Navigator Connect*.

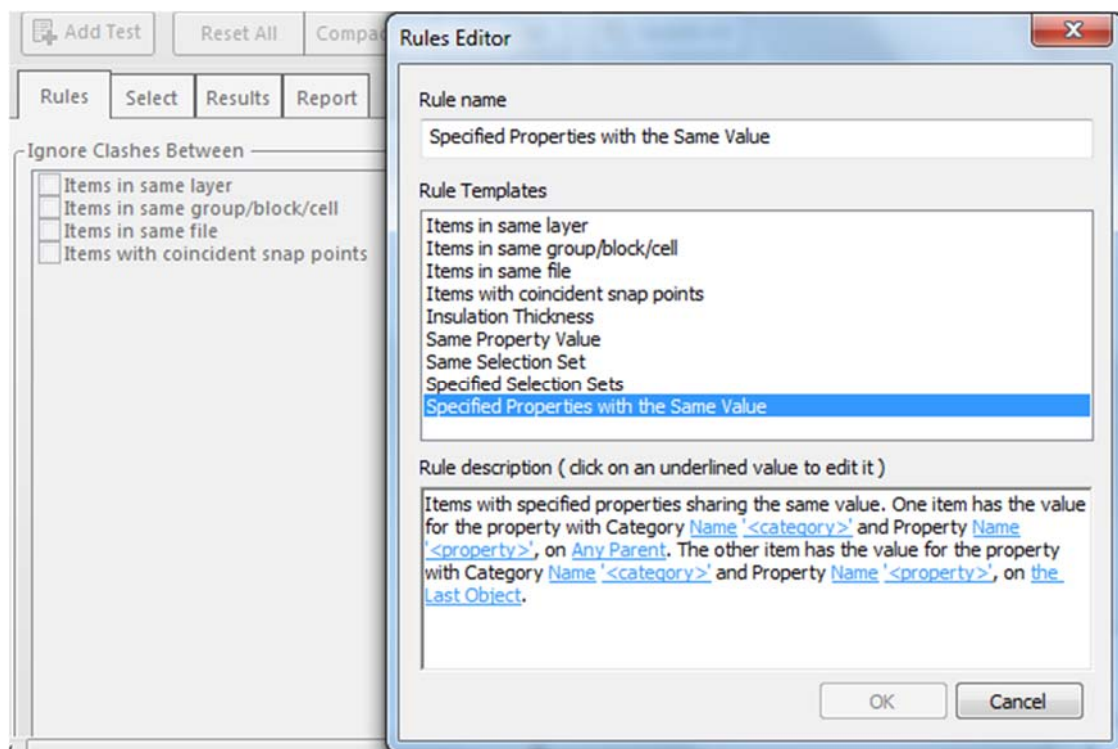
### 2.4.1 Navisworks Manage

- Vývojář: *Autodesk, Inc.*
- Použitá verze: 2017

V současné době je *Navisworks Manage* v České republice jedním z nejpoužívanějších programů, které se zabývají detekcí kolizí. Určitý vliv na tuto skutečnost by mohlo mít značné používání produktů společnosti *Autodesk* na našem trhu., ať už jde o *AutoCAD*, primárně zaměřený na tvorbu 2D dokumentace, nebo již zmíněný 3D nástroj *Revit Architecture*. Díky tomuto faktu má *Navisworks Manage* velikou výhodu oproti ostatním programům, neboť těží z vysoké kompatibility datových formátů. Vývojáři *Autodesku* mají jistou možnost vytvoření monopolu díky integraci převodníků mezi svými programy, které v současnosti pokrývají čím dál více technických oborů.

*Navisworks Manage* nabízí širokou škálu nástrojů pro práci s BIM modelem. Je otázkou, do jaké míry lze jednotlivé funkce efektivně využívat, či zda by nebylo lepším řešením je zahrnout do jiného programu. Jako příklad lze uvést tvorbu animovaných průletů stavbou či renderovací *engine* pro tvorbu vizualizací. Tuto funkci má již *Revit Architecture*.

Co se týče práce s pravidly pro detekci kolizí, nabízí *Navisworks Manage* poměrně pokročilé možnosti jejich programování. Lze je nastavovat v rámci přednastavených pravidel, u kterých lze upravovat konkrétní atributy (obrázek 14). Principem je následné ignorování kolizí, vymezených nastavenými pravidly.



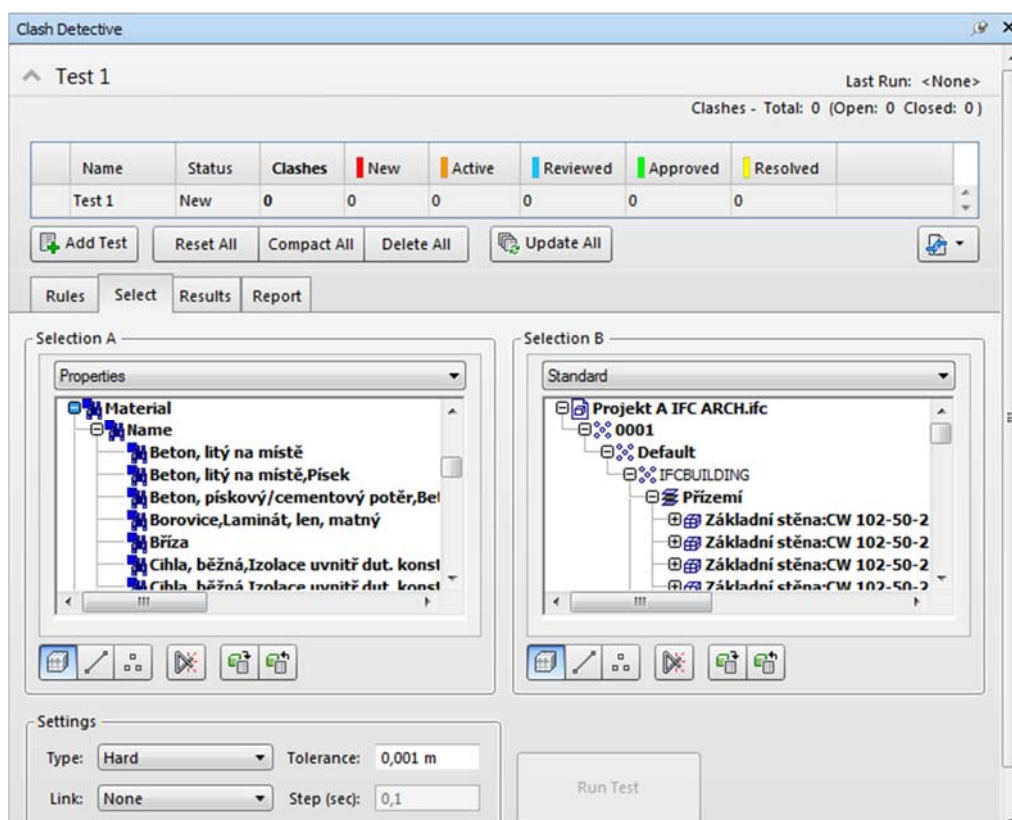
**Obrázek 14: Ukázka nastavení atributů pravidel**

Zdroj: vlastní tvorba s využitím Navisworks Manage

Obecně je zadávání kolizí nastaveno tak, že uživatel definuje ve výběrech A a B objekty, které se mezi sebou mají porovnat (obrázek 15). Tím pádem je možné porovnat celé systémy v budově mezi sebou, či se zaměřit jen na komponenty rozdělené po patrech. Tato vlastnost však závisí na struktuře importovaného modelu. V případě, že je nahrán IFC model stavby, ve kterém nejsou definovaná patra, či přehledně označené systémy (TZB, statika, atd.), není možné plně využít možností, které *Navisworks Manage* v tomto ohledu nabízí. Program nabízí tři základní zobrazení pro označení výběrů A a B:

- Standardní
- Kompaktní
- Dle vlastností modelu

Standardním zobrazením se rozumí rozdělení konstrukcí po předem definovaných patrech. V praxi to tedy znamená, že lze vybírat například ze stěn v prvním patře. Kompaktní zobrazení zahrnuje kompletně celý importovaný objekt. A volba dle vlastností modelů nabízí rozdělení dle detailních informací a hodnot, uvedených v rámci exportu do IFC formátu. To působí na první pohled značně nepřehledně, což může být důvod, proč používat tuto volbu pouze ve speciálních případech (vyhledávání objektu dle jeho ID, použitého stavebního materiálu, IFC názvu nebo jiných vlastností).



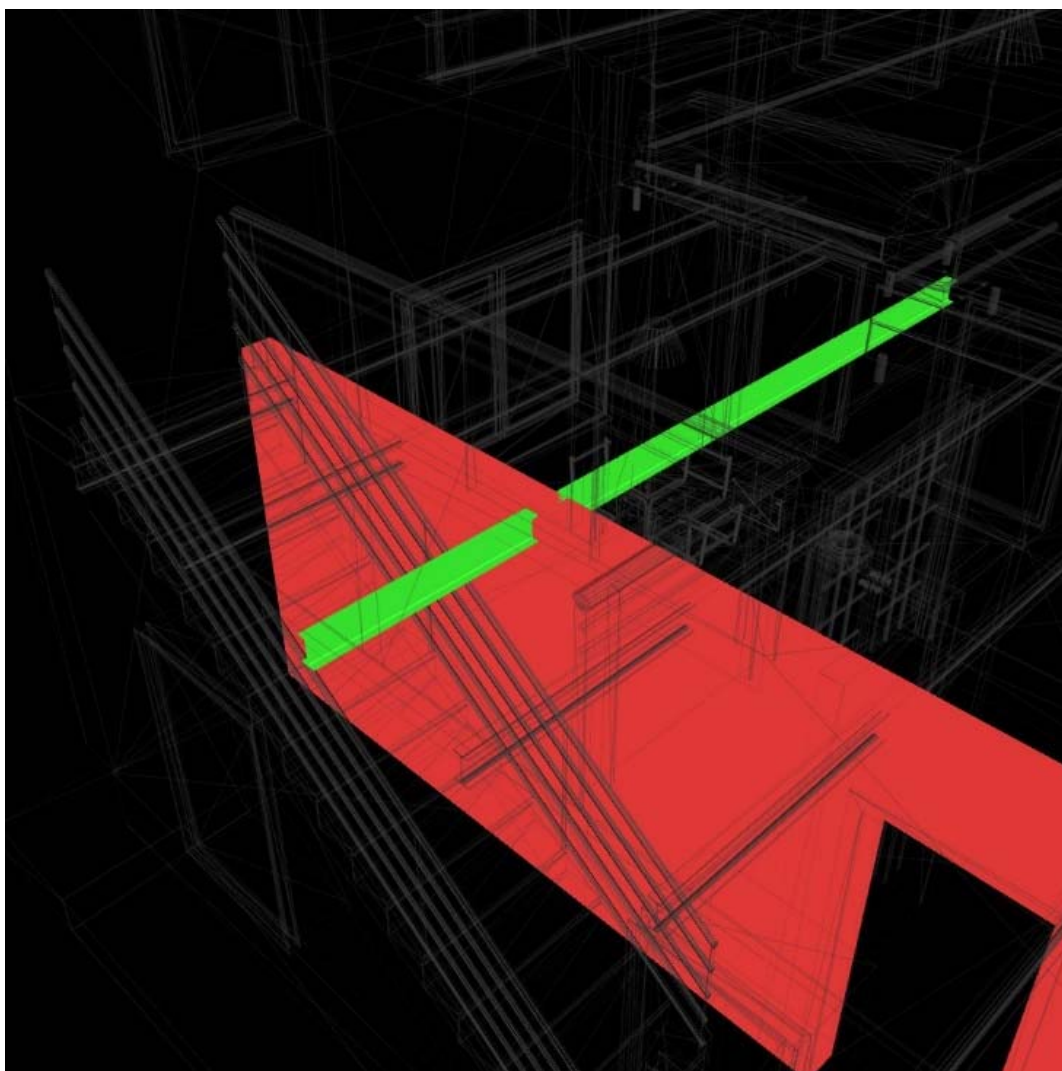
Obrázek 15: Ukázka uživatelského rozhraní nastavení detekce kolizí v programu Navisworks Manage

Zdroj: vlastní tvorba za použití Navisworks Manage

Mimo klasické kontroly kolizí mezi povrchy objektů je také možné zahrnout do detekce další geometrické části, jako jsou body nebo hrany objektů. Ty lze také vybrat zvlášť a ignorovat tak kolize povrchů<sup>6</sup>. Další možností je potom kontrola objektu sebe samého. To usnadňuje výběr systémů. V praxi se totiž nemusí zahrnout stejný systém na obě porovnávané strany, ale stačí zaškrtnout samotnou kontrolu onoho systému. Jiná funkce zase nabízí zahrnout do detekčního výběru pouze to, co si uživatel sám označí v zobrazeném 3D modelu.

<sup>6</sup> Dle mého názoru se však jedná o téměř zbytečnou funkci. Ve většině případů uživatele zajímá hlavně kolize povrchů.

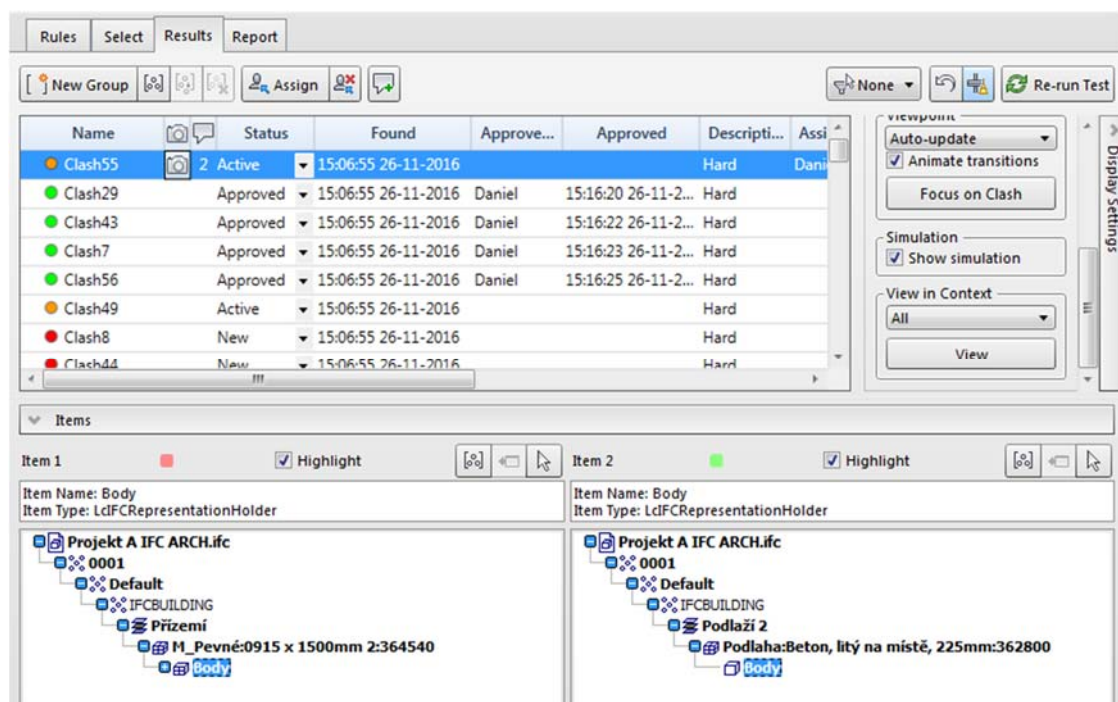
Samotné programování kolizí je zprostředkováno pomocí výběru typů kolizí: *Hard*, *Hard (Conservative)*, *Soft* a *Duplicates*. Kolize typu *Hard* a *Soft* se prakticky shodují s popisem tvrdých a měkkých kolizí v dřívějších kapitolách (kapitola 1.3.1). Typ *Duplicates* kontroluje výskyt zdvojených komponent. Oproti *Hard* kolizi se *Hard (Conservative)* liší v nastavení tzv. průnikové metody (z angl. *Intersection Method*) [22]. Ta se obecně zabývá kontrolou průniků povrchů komponent tvořených malými trojúhelníky, které jsou základním geometrickým prvkem. U *Hard (Conservative)* kolizí jsou však zahrnuty i takové konflikty, při kterých se trojúhelníky neprotínají, ale může to reálně hrozit. To zajišťuje podrobnější kontrolu i za cenu falešných výsledků testu. [22]



**Obrázek 16: Ukázka znázornění kolize programem Navisworks Manage**

Zdroj: vlastní tvorba s využitím Navisworks Manage

S již vyhledanými kolizemi pracuje Navisworks Manage formou označování jejich stavů, delegací řešitelů nebo jejich slučování do složek dle preferencí uživatele. Program danou kolizi zobrazí a vypíše dotčené komponenty (obrázek 17). Po revidování lze exportovat report mj. do HTML formátu (obrázek 18).



Obrázek 17: Zobrazení vyhledaných kolizí v rozhraní Navisworks Manage

Zdroj: vlastní tvorba s využitím Navisworks Manage

Autodesk Navisworks®		Clash Report													
Test 1	Tolerance	Clashes	New	Active	Reviewed	Approved	Resolved	Type	Status						
	0.001m	62	56	2	0	4	0	Hard	OK						
Image	Clash Name	Status	Distance	Description	Date Found	Assigned To	Date Approved	Approved By	Clash Point	Item 1			Item 2		
										Layer	Item Name	Item Type	Layer	Item Name	Item Type
	Clash55	Active	-0.012	Hard	2016/11/26 14:06:55	Daniel Šabart			x:-4.744, y:-0.314, z:2.694	Přízemí	Body	Composite Part	Podlaží 2	Body	LclFCRepresentationHolder
	Clash29	Approved	-0.094	Hard	2016/11/26 14:06:55		2016/11/26 14:16:20	Daniel	x:-5.259, y:-0.314, z:2.581	Přízemí	#1593	IFCBOOLEANCLIPPINGRESULT	Podlaží 2	#150	IFCEXTRUDEDAREASOLID
	Clash43	Approved	-0.061	Hard	2016/11/26 14:06:55		2016/11/26 14:16:22	Daniel	x:-6.239, y:-0.100, z:2.437	Přízemí	#5301	IFCBOOLEANCLIPPINGRESULT	Podlaží 2	#223	IFCEXTRUDEDAREASOLID
	Clash7	Approved	-0.163	Hard	2016/11/26 14:06:55		2016/11/26 14:16:23	Daniel	x:-6.214, y:-0.314, z:2.437	Přízemí	#1593	IFCBOOLEANCLIPPINGRESULT	Podlaží 2	#223	IFCEXTRUDEDAREASOLID

Obrázek 18: Ukázka reportu kolizí z Navisworks Manage ve formátu HTML

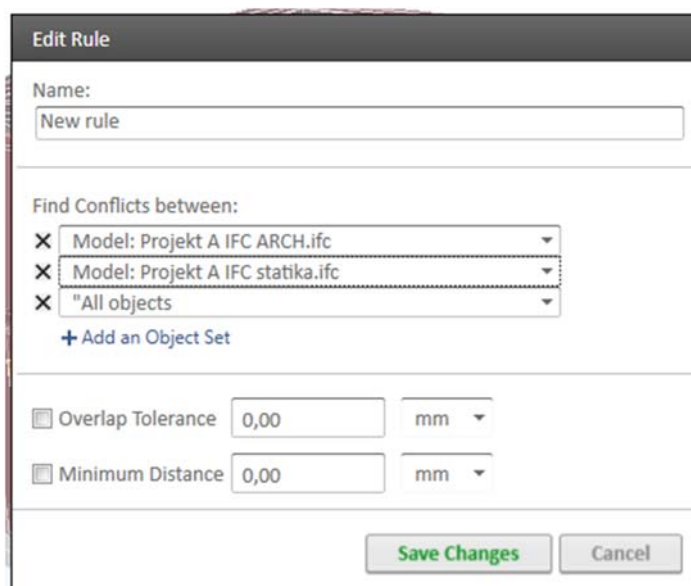
Zdroj: vlastní tvorba s využitím Navisworks Manage

## 2.4.2 Tekla BIMSight

- Vývojář: *Tekla, Trimble*
- Použitá verze: 1.9.7

Předností softwarového nástroje Tekla BIMSight jsou především nulové náklady na jeho pořízení. Společně s kompatibilitou s datovým formátem IFC se jedná o poměrně značná pozitiva pro koncového uživatele. Při importu je možné do modelu snadno přidat i výkresy ve formátu DWG, které ač nejsou zahrnuty do detekčního procesu, nabízí zlepšení orientace. Lze také nahrát i 3D objekty vytvořené v programu SketchUp. Ty nepatří mezi informační modely, neboť v nich nejsou obsaženy doplňující informace. Díky těmto objektům je ale možné obohatit stávající BIM model o unikátní prvky, které nejsou možné vymodelovat v programech k projektování, neboť je zapotřebí značně komplikovaného postupu. Konkrétně může jít například o modely jeřábů, zařízení staveniště, vozidel, zeleně nebo pouze o ohraničení objemů, představujících objekty širších vztahů s okolím.

Programování pravidel pro vyhledávání kolizí je v programu Tekla BIMSight značně omezené. Právě tento nedostatek je pravděpodobně největším dopadem bezplatné licence softwaru. Možnosti programování pravidel vyžadují sestavení mnohdy velice složitých algoritmů, které se u jiných softwarů výrazně podepisují na výsledné pořizovací ceně. Jedinou možnou úpravou pravidel, kterou Tekla BIMSight nabízí, je nastavení minimální vzdálenosti mezi objekty a tolerance přesahu, viz obrázek (obrázek 19).



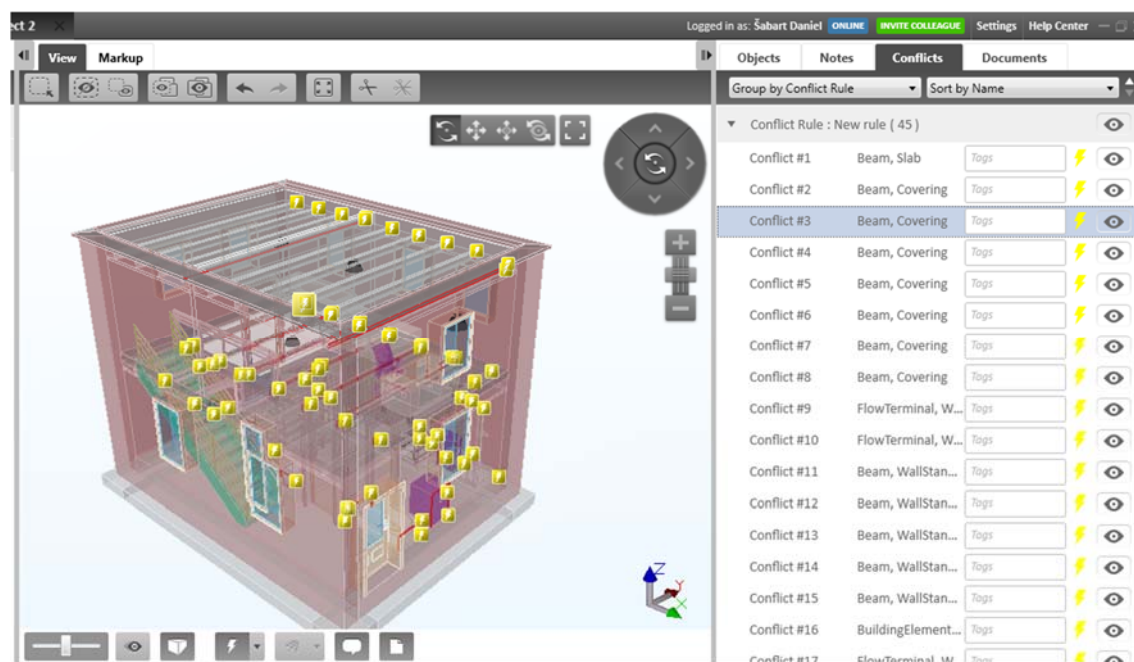
**Obrázek 19: Ukázka možností nastavení pravidel v Tekla BIMSight**

Zdroj: vlastní tvorba s využitím Tekla BIMSight



Princip výběru porovnávaných objektů je z části stejný jako u Navisworks Manage. Lze však označit pouze jednotlivé importované IFC soubory, viz obrázek (obrázek 19). Pokud není uživateli umožněno vybrat jednotlivé části, i třeba dle definic IFC vlastností, je odkázán na správnou spolupráci mezi profesemi. Je totiž potřeba, aby každá profese exportovala svůj vlastní IFC soubor. Další nevýhodou je absence označení modelu jako statický, TZB, atp. Tekla BIMSight nabízí jen označení modelů barvami. To sice může částečně pomoci při orientaci, nikoliv ale při určování kolizních pravidel<sup>7</sup>.

Po spuštění detekce byly vyhledané kolize označeny jak v modelu, tak v podobě seznamu, viz obrázek (obrázek 20). V modelu jsou konflikty zobrazeny pomocí obdélníků s piktogramem blesku v barvě, která náleží nastavenému stavu kolize<sup>8</sup>. Tento stav může souviset jak průběhem řešení kolize nebo její závažností. Problém může nastat v případě, že se bude v modelu nacházet velké množství kolizí. I přes možnost skrytí se může snadno stát, že piktogramy zakryjí velkou část samotného modelu a ztíží tak uživateli práci. Kolize lze třídit dle jejich názvu, označení, tagů, inkriminovaných objektů nebo zadaných pravidel.



Obrázek 20: Ukázka označování kolizí v programu Tekla BIMSight

Zdroj: vlastní tvorba s využitím Tekla BIMSight

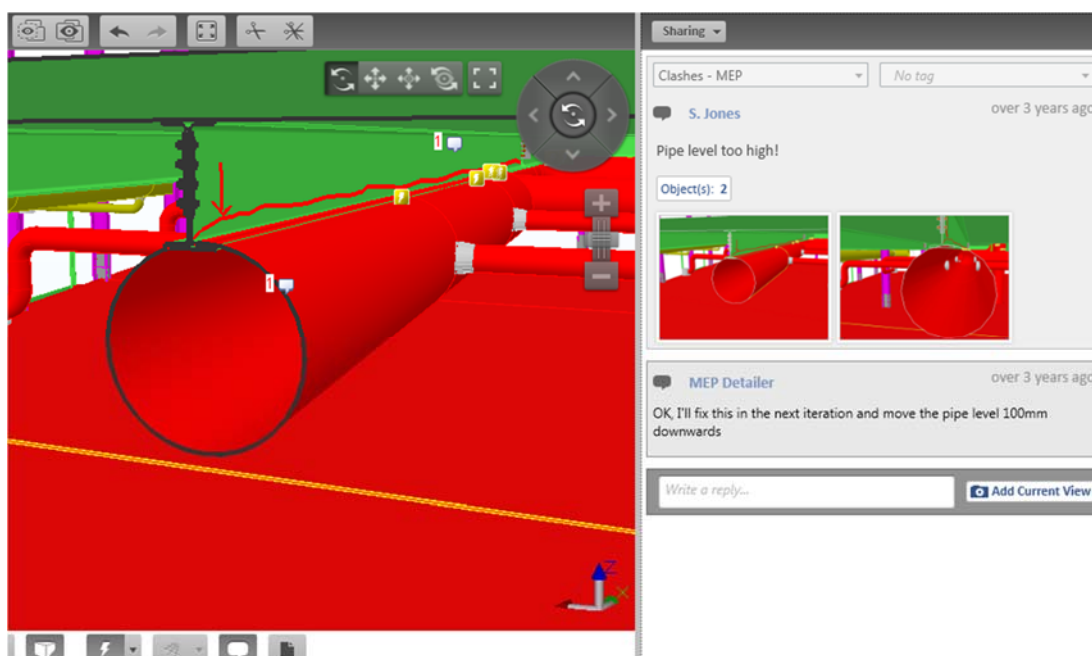
<sup>7</sup> Dle mého názoru se jedná o téměř zbytečnou funkci. Během práce jsem ji vůbec nevyužil.

<sup>8</sup> Tento způsob zobrazování konfliktů nepovažuji kvůli značné nepřehlednosti za ideální.

Během práce s kolizemi je možné pouze zaměřit pohled kamery na vybranou kolizi. To je na rozdíl oproti ostatním programům nevýhoda, neboť ty dokáží ostatní konstrukce zprůhlednit, případně zcela skrýt. Podobný výsledek lze dosáhnout vyhledáním dotčeného objektu v seznamu prvků a manuálně ho izolovat. Problémem je však zafixovaný pohled i po vybrání další kolize. V praxi to tak znamená, že uživatel sice může izolovat jeden objekt, ale při postupu k další kolizi musí opět zapínat zobrazení všech prvků. Je tedy možné dosáhnout stejných výsledků, ale za cenu většího úsilí.

Největším nedostatkem z hlediska práce s kolizemi je absence popisu konfliktu. Tekla BIMSight sice dokáže alokovat konflikt, neuvádí už ale informace o jeho příčině. Oproti programům jako například Solibri se jedná o zásadní nevýhodu. Uživatel tak není schopen přesně určit původ kolize a tudíž je pro něj mnohdy velice obtížné ji začít řešit.

Oproti některým programům na detekci kolizí je Tekla BIMSight vybavena nástrojem odměřování vzdáleností. Dále nabízí připojení různých dokumentů jako například výkresů, detailů, poznámek, tabulek, atp. Zajímavou funkcí je přidávání poznámek a komentářů k jednotlivým kolizím. Lze tak vytvořit jednoduchou konverzaci mezi BIM koordinátorem a týmem řešitelů či projektanty. Neboť software nedisponuje žádným standardním formátem exportu, jedná se o hlavní komunikační kanál mezi zainteresovanými stranami, ukázka viz obrázek (obrázek 21). Program také nabízí možnost kreslit do výkresů obrazce a přidávat komentáře. Vývojáři společnosti Tekla spolupracovali se Solibri také na rozvoji formátu BCF, který zahrnuli i do Tekla BIMSight.



Obrázek 21: Ukázka konverzačního kanálu v podání Tekla BIMSight

Zdroj: vlastní tvorba s využitím Tekla BIMSight



Během importování IFC modelů do Tekla BIMsight se ukázalo, že jistým prvkům byly určeny nesprávné kategorie. Osvětlovací předměty byly zahrnuty do stejné kategorie jako instalační zařízení typu WC nebo sprchový kout. Je otázkou do jaké míry byly na vině nedokonalé nastavené vlastnosti IFC souborů, exportovaných z *Revit Architecture*, nebo chybné převodní procesy v rámci nahrávání do Tekla BIMsight<sup>9</sup>.

### 2.4.3 Solibri Model Checker

- Vývojář: *Solibri*
- Použitá verze: 9.6

Solibri Model Checker je program pro všestrannou kontrolu BIM modelů. Jedná se o specializovaný nástroj na detekci kolizí a komplexní analýzu modelu stavby. Kompatibilita je zajištěna plnou podporou formátu IFC. Je možné aktualizovat již nahrané soubory, což napomáhá ke snížení datových nároků. Za tímto účelem je obsažena i funkce načítání zkomprimovaných IFC souborů bez jejich předchozího rozbalování. To umožňuje ušetření až 80% diskového prostoru. [23] Solibri Model Checker zároveň umožňuje ukládání nahraných IFC souborů do nativního projektového formátu SMC. Takové sloučení jednotlivých souborů opět zmenšuje celkový datový objem zhruba na polovinu.

Kompatibilita s projekčními programy je zajištěna jak IFC formátem, tak doplňky, které se instalují do požadovaných softwarů. Jedním z nich je *add-on* pro *ArchiCAD*, který zajišťuje plné integrování funkcí Solibri Model Checker do projekčního *work-flow*. Prakticky se tak uživatel jedním kliknutím dostane z projektování do procesu kontroly a zpět. [24]

Na podobném principu funguje použití BCF formátu. Uživatel může označit kolizi a připsat k ní komentář ohledně jejího řešení. Následně vše exportuje do souboru s příponou BCF a odešle projektantovi, či v ideálním případě uloží do *cloudu*. Projektant si může soubor importovat do projekčního softwaru (například *Revit Architecture*) a přímo tak načíst informace, které souvisí s vyhledanou kolizí. Zobrazí se mu tak přímo místo kolize, související komentář a vyznačí kolizní prvky. Na rozdíl od výše popisovaných zásuvných modulů se jedná o obecné řešení, které lze využít v různých programech. Základy BCF platformy položily společnosti *Solibri* a *Tekla*. Jejich snahu podpořili vývojáři z firem *Autodesk*, *DDS*, *Eurostep*, *Gehry Technologies*, *Kymdata*, *MAP*, *Progman* a *Quick Pen International*. [15] To samo o sobě svědčí o slibné budoucnosti této formy komunikace.

---

<sup>9</sup> I přes značnou nejistotu jsem se nevyhnul pochybám o schopnostech softwaru Tekla BIMsight v otázce správného importování IFC souborů.



Po nahrání modelů do Solibri Model Checker je třeba určit jejich disciplíny. Těmi se v případě tohoto softwaru rozumí typy modelů jako: architektonický, statický, vedení elektra, vzduchotechniky, potrubí, a mnoho dalších. Solibri Model Checker nabízí nastavení automatického přiřazení těchto disciplín, pokud je při importu zahrnuta informace o zdrojovém programu. Ve výsledku je tak možné přiřadit například modelu z *ArchiCAD* automatické zařazení jako architektonického, Revit MEP jako model TZB, atp.

Software automaticky přiřazuje konkrétní kategorie všem objektům na základě informací vyexportovaných z modelářského programu. Mnohdy je však nutné zařadit některé z prvků ručně. Nejen pro tento účel využívá Solibri Model Checker funkci *To-do list*. Ta uživateli zobrazí, které kroky je třeba udělat, aby bylo možné s modelem správně pracovat.

Hlavní předností softwaru je detailní nastavení pravidel pro detekci kolizí. Solibri Model Checker nabízí rozsáhlou knihovnu pravidel již při instalaci<sup>10</sup>. Ty obsahují velké množství nastavení a obecně hledisek, ze kterých se dají modely kontrolovat. Předinstalovaná pravidla slouží jako výchozí bod pro tvorbu uživatelsky definovaných pravidel. V praxi to znamená, že si uživatel může zkopírovat již vytvořené pravidlo a následně ho přejmenovat a upravit zcela dle svých potřeb<sup>11</sup>. Výhodou je velice snadná správa a manipulace s pravidly. Ty jsou ukládány do souborů ve formátu CSET a lze je tak jednoduše kopírovat a šířit mezi uživateli.

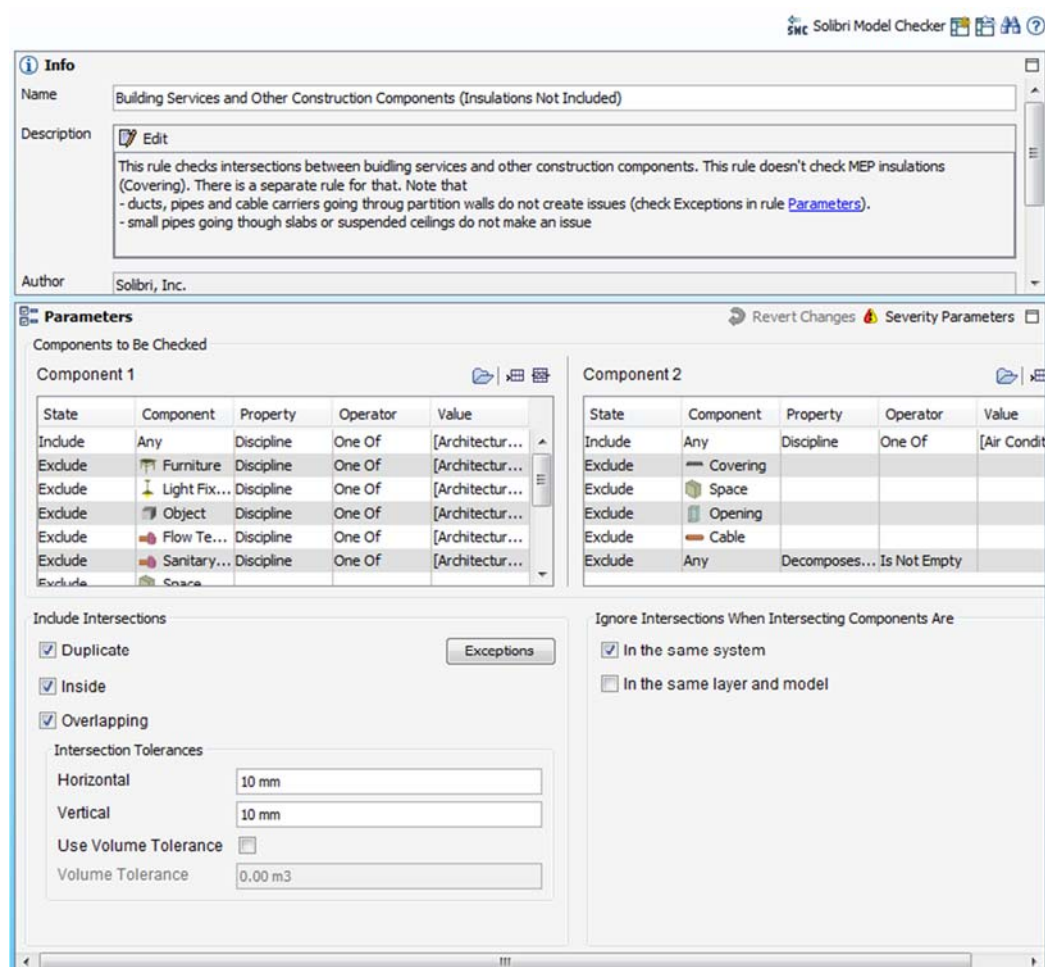
*Solibri Model Checker umí prověřit BIM projekt z hledisek, jako jsou usnadnění přístupu, kontrola výpustí rozvodů, splnění požadavků národních norem či jiných standardů s ohledem na konkrétní inženýrskou disciplínu, či požadavků investora nebo zúčastněných projekčních firem. Navíc mnoho firem používá Solibri Model Checker pro vytváření a kontrolu dodržování vlastních vnitropodnikových standardů. [23]*

Samotné programování pravidel pro vyhledávání kolizí probíhá v dialogovém okně (obrázek 22), jehož obsah závisí na zvoleném typu výchozího pravidla. Výběrem předlohy tak uživatel určuje výsledné chování programu. Následně je pak možné slučovat pravidla do tzv. sad a třídit je dle požadovaných kategorií či profesí.

---

<sup>10</sup> Předinstalovaná pravidla shledávám jako velmi užitečné řešení. V mnoha případech jsem potřeboval nastavit určité typy pravidel a

<sup>11</sup> Úprava pravidel mi přišla zprvu poměrně komplikovaná. Stačilo se však seznámit s principem programování a práce se značně zjednodušila. Musím dokonce ocenit velmi dobrou variabilitu nastavení.



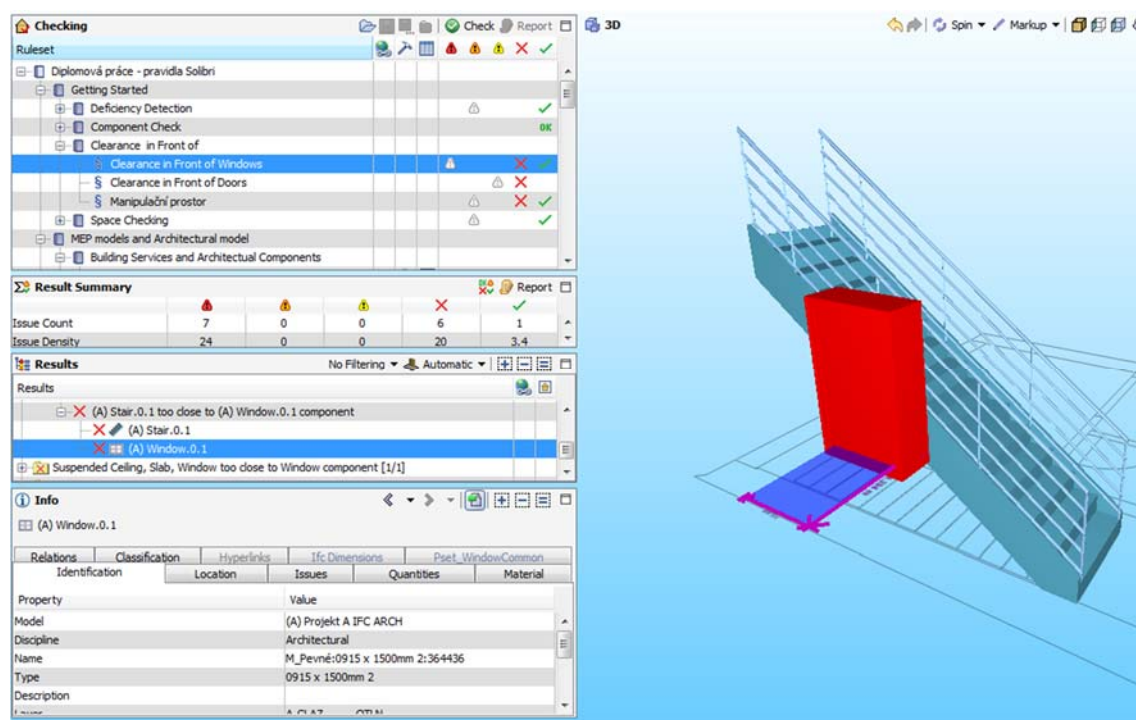
Obrázek 22: Ukázka programování pravidel v programu Solibri Model Checker

Zdroj: vlastní tvorba s využitím Solibri Model Checker

Přednastavená pravidla programu Solibri Model Checker zahrnují nejenom tvrdé a měkké kolize stavebních prvků, ale zabývají se i kontrolou dodržení protipožárních norem, únikových cest z budovy, odpovídajících prostorových nároků pro osoby na invalidním vozíku, apod. Zároveň software upozorňuje na nedostatky v návrhu, jako je chybějící horní a dolní připojení statických prvků na architektonický model, chybějící označení místností či požárních okruhů, nedostatečné vzdálenosti mezi konstrukcemi, apod.<sup>12</sup>. Všechna tato upozornění vychází z určitých obecných pravidel. Ta jsou sice vytvořena vývojáři, přesto je lze poměrně snadno upravit tak, aby byla v souladu s požadavky legislativy, investora nebo dalších zainteresovaných stran.

<sup>12</sup> Většinu z těchto funkcí jsem nevyužil a byly mi spíše na obtíž. Dovedu si však představit jejich značné využití v různých odvětvích stavební produkce, pakliže budou správně použity. Právě v této variabilitě vidím jednu z největších přidaných hodnot Solibri Model Checker.

Solibri Model Checker automaticky klasifikuje závažnost vyhledaných kolizí. Podle Beneše je díky této funkci možné eliminovat lidský faktor chybovosti a odbourat časově náročné vyhodnocování analýz. [23] Software automaticky zařadí kolize do tří druhů podle stupně závažnosti: nízká, střední a kritická. Uživatel pak může kolizi přijmout nebo odmítnout. Tato funkce je spojena s tvorbou reportů a komunikací. Pro lepší přehlednost práce s kolizemi je konflikt vždy po kliknutí zacílen a spolu s inkriminovanými předměty izolován, jak je znázorněno na obrázku (obrázek 23).



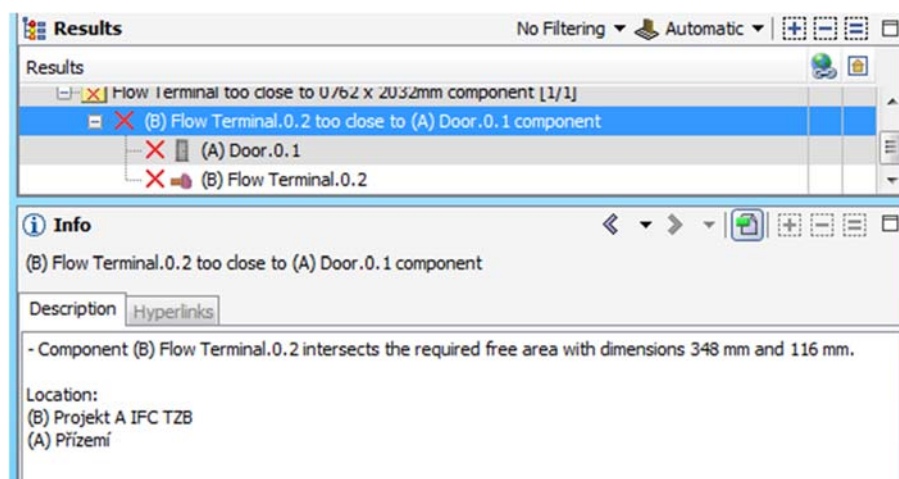
Obrázek 23: Ukázka vyhledané kolize v Solibri Model Checker

Zdroj: vlastní tvorba s využitím Solibri Model Checker

U některých typů kolizí (především měkkých) zobrazuje Solibri Model Checker požadované půdorysné rozměry i přímo v modelu. Ty pak lze exportovat samostatně jako doplňující informace ke kolizím. Během práce s kolizemi je často potřeba skrývat a znovu zobrazovat komponenty. Solibri Model Checker nabízí takovou manipulaci s viditelností objektů na základě začlenění prvků do kategorií. Je tak možné jednoduchým výběrem skrýt například všechny dveře, okna, nosníky a další stavební prvky. Samozřejmostí je pak funkce zprůhlednění konstrukce a vytváření řezů. Zajímavým doplňkem je systém tzv. *Selection basket*, kterým je možné vytvářet různé výběry prvků, jež lze následně ukládat<sup>13</sup>.

<sup>13</sup> Během práce jsem tuto funkci nevyužil a nejsem plně přesvědčen o jejím přínosu.

Jednou z velkých předností programu je detailní označení kolizí. Uživatel se po vybrání konfliktu ze seznamu zobrazí informace o dotčených předmětech, slovní okomentování a umístění kolize, viz obrázek (obrázek 24). Popis je u přednastavených pravidel automaticky generován a zahrnuje i proměnlivé hodnoty. U nově vytvořených pravidel lze tento popis vytvářet. Všechny tyto údaje lze posléze importovat do prezentace ke slajdu s vyobrazenou kolizí. Zároveň je možné kreslit ruční poznámky a přidávat kóty přímo do prostoru modelu.



**Obrázek 24: Ukázka popisu kolize pomocí Solibri Model Checker**

Zdroj: vlastní tvorba s využitím Solibri Model Checker

Solibri Model Checker dokáže využívat informace obsažené v IFC modelech nejenom na detekci kolizí, ale i na tvorbu různých výkazů výměr. Ty je pak možné importovat do specializovaných rozpočtářských programů.

Při řešení kolizí nabízí Solibri Model Checker přímé delegování odpovědného pracovníka. Software nabízí několik typů exportu informací. Lze využít jednoduchou prezentaci s okomentováním jednotlivých kolizí<sup>14</sup>. Lze do něj zahrnout informace o kolizi, kde se nalézá, komu byla přidělena a jaký status v současnosti nabývá. Navíc software umožňuje přidat odkaz na externí soubory jako například technické listy, normy nebo výkresy.

Sofistikovanějším výstupem jsou však reporty ve formátech xls, pdf a rtf, které mají přednastavenou podobu a obsahují větší množství informací. Jejich obsah si uživatel volí přímo při exportu dle označení kolizí. Strukturu reportu lze upravovat do jisté míry přímo v Solibri Model Checker, nebo potom v aplikacích jako Microsoft Excel, Adobe Acrobat, apod. Ukázka standardního reportu je na obrázku (obrázek 25).

<sup>14</sup> Funkce vytváření prezentací mne zaujala. Samotné vytváření slajdů je však dle mého názoru zbytečně zdouhavé a nabízí jistý prostor pro automatizaci.

SOLIBRI		Presentation 1						
Model Name	Projekt A SMC Version: 9.6							
Checker	Bc. Daniel Šabart							
Organization	CTU in Prague							
Date	December 10, 2016							
Projekt A IFC ARCH	Date: 2016-11-22 17:27:42 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3							
Projekt A IFC TZB	Date: 2016-11-22 17:27:25 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3							
Projekt A IFC statika	Date: 2016-11-22 17:27:03 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3							
Presentation 1								
Number	Id	Location	Date	Author	Title	Picture	Issue Description	Issue Comment
1	7	(C) Projekt A IFC statika (A) Přizemí, (A) Podlaží 2	10-Dec-2016	DŠ	Beam, Suspended Ceiling, Light Fixture, Slab too close to Window component		Clearance in Front of Windows	DŠ, 10/12/2016: Okno zasahuje do podlahy 2NP
2	25	(C) Projekt A IFC statika (A) Přizemí, (A) Podlaží 2	10-Dec-2016	DŠ	Okno vs podhled		Okno zasahuje do podhledu místnosti	
3	23	(A) Přizemí	10-Dec-2016	DŠ	Příčka vs okno		Příčka zasahuje do okna	

Obrázek 25: Ukázka reportu pomocí Solibri Model Checker

Zdroj: vlastní tvorba s využitím Solibri Model Checker

#### 2.4.4 BIMReview

- Vývojář: *StruMIS LTD*
- Použitá verze: 8.1.3.0

*BIMReview* nabízí mimo detekce kolizí jen několik málo dalších funkcí. Mezi ně se řadí například odměřování vzdáleností, které je řešeno systémem úchytných bodů, známým například z programu *AutoCAD*<sup>15</sup>. Přidanou hodnotu může *BIMReview* nabídnout ve formě možnosti importu Ganttova diagramu. Je možné tak sledovat různé fáze výstavby a kontrolovat v nich kolize. Teorie detekce kolizí však zastává obecný názor, že by se všechny konflikty měly vyřešit co nejdříve. Nabízí se tak otázka, proč by měl uživatel vyžadovat funkci hledání kolizí v jednotlivých fázích projektu, když je má ideálně najít všechny najednou již na začátku.

Jako další doplňkovou funkci můžeme vnímat obarvování, značkování a vyhledávání konstrukcí dle nahraných atributů. Vzhledem k tomu, že se má jednat o soubory ve formátu IFC, jsou do výběru atributů zahrnuty veškeré informace, které IFC obsahuje. Za jistých okolností je tak možné těžit ze systému vyhledávání dle těchto údajů, jež nebývá

<sup>15</sup> Dle mého názoru se však nejedná o dobré řešení. Nedá se tak změřit prakticky žádný rozměr s požadovanou přesností.





v podobných programech zahrnut. S touto funkcí je spojena možnost animace výstavby opět na základě atributů.<sup>16</sup>

Samotnou detekci kolizí řeší *BIMReview* na podobném principu, jako výše popisovaný *Navisworks Manage*. Výběr prvků je však rozdělen dle připojených IFC modelů, ze kterých lze filtrovat komponenty dle IFC parametrů. Vzhledem k tomu, že tyto parametry mají svou specifickou podobu, není tento systém přehledný. Slabinou je také nutnost vícenásobného výběru modelu v případech, kdy uživatel potřebuje vybrat více komponent. To opět značně ubírá na přehlednosti. Pro obecné kontroly kolizí mezi architektonickými, TZB, statickými a dalšími modely je však tato funkce dostačující. Zároveň je i v tomto programu zachována možnost kontroly každého zvoleného modelu jako takového.

*BIMReview* nerozděluje kolize na tvrdé či měkké. Místo toho používá nastavení hodnoty parametrů přesahu (*overlap*) a minimální vzdálenosti. Ty vymezují tolerance pro detekci obou typů kolizí. To však znamená, že není možné použít oba parametry v jedné kolizní úloze. Je potřeba nastavit dvě úlohy – jednu pro tvrdé kolize a druhou pro měkké. Stejným způsobem je třeba postupovat i v *Navisworks Manage*.

Dále je možné nastavit ignorování detekce u zvolených objektů pomocí IFC parametrů ve formě filtrů. Ty se pak dají ukládat a načítat v jiných projektech. Využitelnost je však pouze teoretická, protože jak již bylo zmíněno, stávající parametry IFC jsou poměrně komplikované a málo přehledné. Možnost filtrování je však obecně vhodným doplněním nastavení detekce kolizí.

Po vyhledání kolizí je možné pomocí nástroje izolace zobrazit dotčené komponenty. Dále lze konfliktu přiřadit určitý status v závislosti na stavu jeho řešení. Uživatel může přidat komentáře do poznámek ke kolizi a do dalšího sloupce uvést konkrétního řešitele<sup>17</sup>. Zobrazení kolizí ve 3D modelu je zpracování pomocí barevných šipek. I přes možnost jejich skrytí a zobrazení se tento systém jeví dosti nepřehledným (obrázek 26). Oproti ostatním zkoumaným programům chybí uvedení kolizních prvků pro jejich jednoznačnou identifikaci<sup>18</sup>.

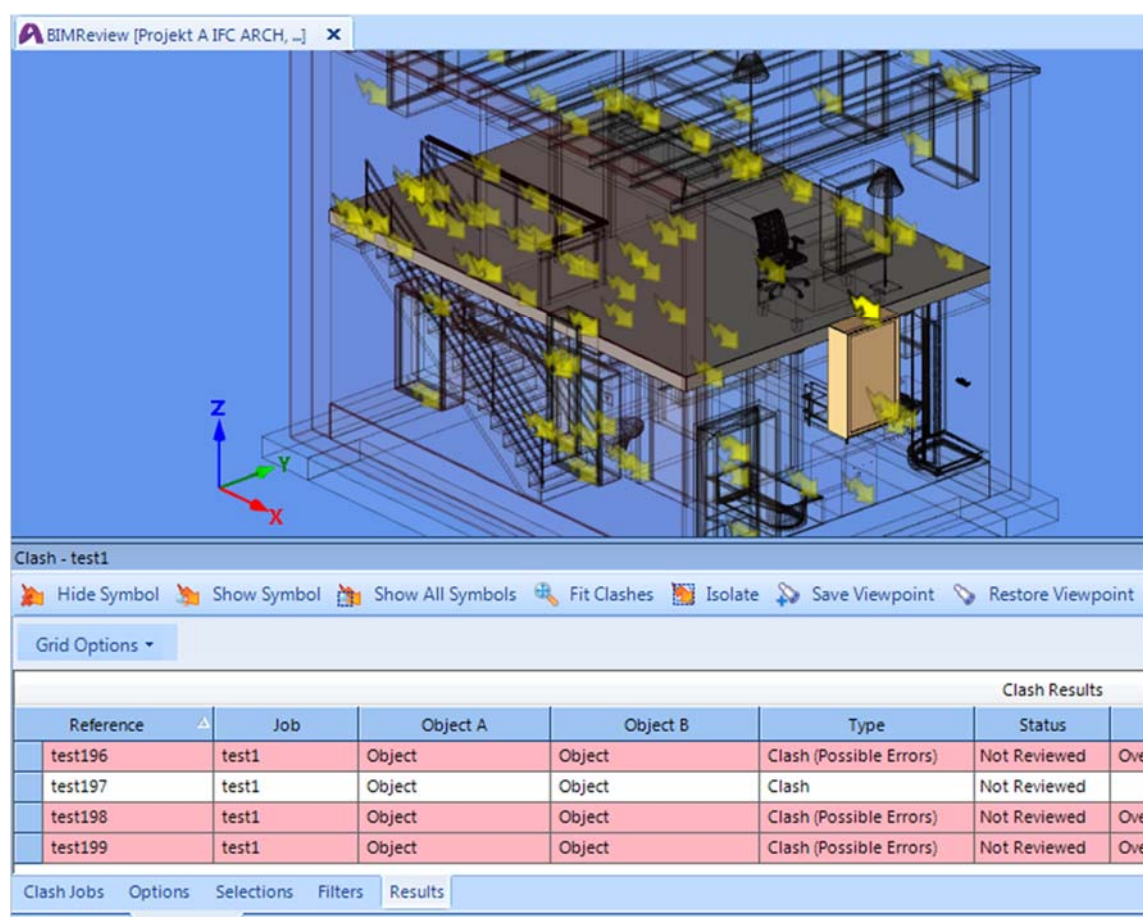
---

<sup>16</sup> Otázkou však je, zda samotná definice IFC komponent stačí k sestavení takto komplikované animace. Lepší zpracování nabízí dříve zmiňovaný *Navisworks Manage*.

<sup>17</sup> Zde jsem nabyl dojmu, že se jedná o dvě zdánlivě různé věci řešené stejným způsobem. V ostatních programech bývá většinou funkce delegování podchycena o poznání lépe.

<sup>18</sup> Tuto absenci řadím mezi hlavní nedostatky programu *BIMReview* z důvodu zásadního významu při identifikaci kolizních prvků i z hlediska následné komunikace.

Přímý výstup detekovaných a revidovaných kolizí je řešen pouze formou obrazové dokumentace. Ty je možné přímo z aplikace odeslat elektronickou poštou. Pro zlepšení spolupráce jsou však zahrnuty další funkce jako systém 3D poznámek, dotazů či grafických značek. Ty je možné libovolně vkládat do modelu s návazností na vyhledané kolize. Podporou pro koordinaci je pak možnost komentářů jednotlivých poznámek.



Obrázek 26: Ukázka přehledu kolizí a jejich znázornění v modelu pomocí BIMReview

Zdroj: vlastní tvorba s využitím BIMReview

Praktickým nástrojem je tzv. *BIM Share*. Ten nabízí pokročilé možnosti při poměrně jednoduchém sdílení dat. Principem je uložení informací generovaných programem do nově vytvořené složky, která je sdílena více uživateli. V manuálu programu je přímo uveden postup zahrnující sdílenou složku aplikace *Dropbox*. Velkou výhodou je konzistentní hierarchie podsložek. Ta může velice usnadnit orientaci v podkladech a nabízí možnost sdílení výstupů *BIMReview* klasickou cestou pomocí systémového prohlížeče či obecně používaných nástrojů.



## 2.5 Vyhodnocení vlastností porovnávaných softwarů

V rámci celé této kapitoly (kapitola 2) bylo zapotřebí určit jeden program pro následné použití při praktických úlohách. Vybraný software měl mít nejvhodnější nástroje pro detekci kolizí a reprezentovat tak rodinu programů, k tomuto účelu určených.

Základní kritéria, popsaná v jedné z dřívějších kapitol (kapitola 2.1), byla splněna již při výběru softwarů do přehledové tabulky (tabulka 1). Z té vyplývá, že vybrané programy mají z velké části srovnatelné obecné vlastnosti. Podrobnějším rozбором a zkoumáním práce v programech se však ukázalo, že ač nabízí stejné funkce, v některých z nich jsou jejich možnosti rozdílné.

Konkrétním příkladem je rozdíl v možnostech programování pravidel pro vyhledávání kolizí pomocí nástrojů Tekla BIMsight nebo BIMReview a Solibri Model Checker. Zatímco dva prvně jmenované softwary neposkytují téměř žádný prostor k modifikaci pravidel, Solibri Model Checker nabízí velmi rozsáhlé možnosti úprav. Navisworks Manage v tomto ohledu sice nedosahuje takových kvalit, poskytuje však obecně lepší funkční zázemí, než Tekla BIMsight a BIMReview.

Pro účely této práce byl primární výběr programů zaměřen na takové nástroje, které se specializují na detekci kolizí v BIM modelech staveb. Během průzkumu Navisworks Manage byly z tohoto důvodu některé jeho funkce označeny za přebytkové. Specializovanějším programem se tak jeví Solibri Model Checker, u kterého nebyly nalezeny žádné podobné přebytky. Svým zaměřením přesně odpovídá požadavkům této práce. To je podpořeno i názory z praxe, kdy je tento nástroj mnohdy zmiňován jako jeden z nejlepších při debatách o problematice vyhledávání kolizí.

Při výběru ideálního programu na detekci kolizí pro stavební nebo jinou (například developerskou, projekční, atd.) společnost, by byla důležitým kritériem cena. Z tohoto hlediska by byl nejvhodnějším nástrojem Tekla BIMsight, který je nabízen zdarma. Nejdražší zkoumanou variantou je pak Solibri Model Checker. Dále mohou být společnosti často ovlivněny již používaným typem či výrobcem programů a podle toho volit i další software.

Výše uvedené faktory však byly v této práci potlačeny kvůli zachování objektivního přístupu ke zkoumání samotných vlastností programů. Na základě vynikajících schopností úpravy pravidel pro vyhledávání kolizí, kladných ohlasů veřejnosti a v neposlední řadě subjektivním dojmu autora během zkoumání, byl pro další část této práce vybrán nástroj Solibri Model Checker.



### 3 Vyhodnocení přínosu využití BIM pro detekci kolizí

Hlavním cílem diplomové práce je vyhodnocení přínosu detekce kolizí s využitím BIM oproti klasickému postupu bez využití výpočetní techniky a specializovaných softwarů. Aby bylo možné porovnávat efektivitu práce v těchto rozdílných rovinách, bylo nutností vytvořit modelové úlohy. Ty musely být přímo uzpůsobeny zkoumané problematice a zároveň splňovat určitá opatření, která zajistila jejich použitelnost pro řešení více řešiteli za relativně krátkou dobu. Následně byl sestaven výpočet pomocí několika rovnic, kterým lze po dosažení zjištěných hodnot určit poměr mezi jednotlivými metodami vyhledávání kolizí. Rovnice jsou popsány v další části této práce (kapitola 4). Výsledky pak číselně vyjadřují, do jaké míry je využití BIM nástrojů při vyhledávání konfliktů v projektu efektivnější než klasický postup. Zároveň lze po úpravě použít tuto metodiku i pro porovnání účinnosti dvou programů.

#### 3.1 Typy úloh a způsob řešení

Pro účely výzkumu byly sestaveny dva jednoduché projekty dvoupodlažních objektů. V rovnicích jsou označeny obecným indexem  $i$ , který pro tento případ nabývá hodnot  $X$  a  $Y$ . Tyto projekty byly vymodelovány v programu *Revit Architecture* kvůli požadavku na následné řešení pomocí vybraného BIM nástroje na vyhledávání kolizí. Každý projekt byl vytvořen se záměrnými chybami v návrhu, viz jedna z následujících podkapitol (kapitola 3.2). Zároveň byly vygenerovány výkresy vždy jednoho půdorysu prvního podlaží a dvou na sebe kolmých řezů pro každý z projektů. Následně byly použity jako podklad pro řešení bez výpočetní techniky a BIM softwaru. Tyto výkresy jsou uvedeny v přílohách této práce.

Aby byla co nejlépe zachycena činnost řešitelů při vyhledávání kolizí na obou projektech, byly vytvořeny dva typy úloh. Postup byl u obou projektů stejný. V první úloze dostali řešitelé za úkol označit na výkresech, co největší počet kolizí a zároveň byl nastaven časový limit na dvě minuty. Tato úloha na rychlost nese v rovnicích označení  $Q$  (z anglického *quick*). Druhá úloha měla pro řešitele stejný cíl – tedy nalézt co nejvíce konfliktů, avšak tentokrát se časový limit zvýšil na pět minut. Zároveň nemuseli označovat již vyhledané kolize z úlohy  $Q$ . Tato úloha na přesnost je v rovnicích označena písmenem  $A$  (z anglického *accurate*). Smyslem rozdělení na dvě úlohy byla snaha zachytit zároveň rychlost i přesnost práce řešitelů. Naměřené údaje byly následně dosazeny do rovnic a použity při výpočtu. Nutností však bylo řešitele uvést do problematiky a podrobně vysvětlit zadání úloh. Toto je popsáno v jedné z následujících kapitol (kapitola 3.3).



V rámci porovnání byly detekce kolizí v obou projektech řešeny i pomocí softwaru. Ten byl určen na základě průzkumu v předchozí části této práce (kapitola 2). Vybraným nástrojem se stal Solibri Model Checker. Zde již nebylo použito rozdělení na přesnou a rychlou úlohu. Program měl za úkol nalézt všechny kolize v projektu a byla zaznamenána doba, po kterou je vyhledával. Příprava programu a postup úloh je uveden v jedné z následujících kapitol (kapitola 3.4).

Tímto způsobem bylo možné získat informace o efektivnosti člověka a softwaru jako řešitelů. Pomocí rovnic pak bylo možné tyto dva odlišné postupy porovnat.

### 3.2 Tvorba BIM modelů projektů

Při vytváření informačních modelů bylo třeba se řídit určitými zásadami korektního modelování kvůli dodržení jisté kvality projektu. To například znamená správné napojování konstrukcí, zadávání parametrů stěn a obecných vlastností konstrukčních prvků. Během práce však byla tato pravidla vícekrát záměrně porušena, aby vznikly kýžené kolize. V případě, že by bylo dodrženo vše tak, jak má být, nebyly by projekty použitelné. Ani jeden z řešitelů (lidé i software) by nenašel byť jedinou kolizi. Projekty byly vytvořeny s odlišným počtem kolizí. První projekt (X) obsahoval celkem 45 kolizí, kdežto v druhém projektu (Y) jich bylo na 127. Tyto počty byly stanoveny programem Solibri Model Checker a pro účely této práce jsou považovány za konečné. V případě, kdy program neurčí všechny kolize, bylo by třeba upravit postup výpočtu, jak je zmíněno v další části práce (kapitola 4.1.1).

Je důležité si uvědomit, že některé kolize není možné s omezenými informacemi v podobě pouhých tří výkresů ručně nalézt. V praxi by bylo nejspíše k dispozici více výkresů. Otázkou je, zda by větší počet mnohdy velmi složitých výkresů nebyl spíše matoucí a nestěžoval by vyhledávání konfliktů. Zároveň by se v projektu vyskytoval mnohem vyšší počet kolizí. V reálných projektech (obzvláště velkých administrativních budov) se často pohybuje v řádech tisíců. Z důvodu méně komplikované přípravy podkladů pro praktické úlohy byly vytvořeny projekty značně menších rozměrů. Počet kolizí v nich obsažených se pohyboval v řádu desítek.

Společně se správným informačním modelováním bylo třeba dodržet správný typ exportního datového formátu. IFC má v dnešní době více variant, které se liší mimo jiné i definicí prvků. V rámci této práce byla použita verze *IFC 2x3*. Ta byla pro své starší datum vydání zvolena pro zaručení kompatibility v každém ze softwarů. V rámci exportu byly vytvořeny IFC soubory pro jednotlivé profese – architektonický návrh, statické prvky a TZB.



Zároveň bylo nastavení vlastností výstupního souboru přizpůsobeno práci s programy na detekci kolizí. Proběhl i export do formátu i.dgn pro práci v *Bentley Navigator Connect*, pomocí speciálního zásuvného modulu přímo v *Revit Architecture*. Bohužel se však nepodařilo jej načíst do *Bentley Navigator Connect*.

### 3.3 Zpracování úloh řešiteli

Pro získání relevantního vzorku člověkem zpracovaných úloh bylo třeba nashromáždit dostatečně velkou skupinu řešitelů na stejné znalostní úrovni. V ideálním případě by realitě nejvíce odpovídal vzorek profesních expertů, kteří se zabývají manipulací a kontrolou BIM modelů. Toho však nebylo možné z technických a organizačních důvodů dosáhnout. Žádný smysl by ale nemělo řešení úloh laickou veřejností. Lidé bez technického a zejména pak stavebního vzdělání by nemohli tvořit reprezentativní vzorek člověka jako řešitele.

Do role řešitelů byli vybráni studenti vyšších ročníků stavebních fakult technických vysokých škol. Toto rozhodnutí bylo ovlivněno, kromě výše uvedených důvodů i faktem, že skupina studentů je časově flexibilnější než jiné vzorky, obzvláště pak vzorek profesních expertů. Svými znalostmi mají navíc předpoklady převyšovat laickou veřejnost a po vysvětlení úloh a postupu řešení se přiblížit činnosti profesních expertů. Tohoto výzkumu se zúčastnili studenti prvního ročníku magisterského studia oboru Projektový management a inženýring Fakulty Stavební, Českého Vysokého Učení Technického v Praze.

Před zahájením samotného řešení úloh bylo zapotřebí nejdříve studentům objasnit základy teorie kolizí. To bylo zajištěno krátkou prezentací s ukázkami a vysvětlením některých typů konfliktů, které se v úlohách mohly vyskytovat. Záměrně byli vybráni studenti předmětu *BIM – Informační modelování staveb*, kteří již měli základní znalosti o tvorbě BIM modelů. Zároveň byla stanovena pravidla pro průběh řešení úloh. Ta obsahovala i použití propisovacích tužek dvou rozdílných barev pro obě úlohy. První barvou měli studenti označit kolize v úloze na rychlost (úloha Q) a následně druhou barvou zakreslit konflikty v úloze na přesnost (úloha A). Tím bylo zajištěno rozeznání při sčítání výsledků a umožněno rozdělení jednotlivých úloh pro následné vyhodnocení. Čas, který zabralo vysvětlování teorie a zadání úloh je zohledněn v rovnicích v obecném koeficientu, viz rovnice (4-1).

Do řešení úloh se zapojilo celkem 22 (respektive 21) studentů. Na projektu X určili řešitelé dohromady 131 kolizí v úloze Q a 213 v úloze A. Po zpracování projektu Y označili 133 konfliktů v úloze Q a 284 v úloze A. Toto jsou vstupní hodnoty do výpočtu relativní účinnosti, která je definována v jedné z následujících kapitol (kapitola 4.1). Podrobné výsledky řešitelů jsou k dispozici k nahlédnutí v přílohách této práce.



### 3.4 Zpracování úloh BIM nástrojem

Pro vyhledání kolizí v modelových projektech pomocí softwaru Solibri Model Checker bylo zapotřebí několika základních kroků nastavení programu. V první řadě bylo nutné po importování jednotlivých IFC souborů určit jejich profesní kategorie. Vzhledem k tomu, že software umožňuje vybrat z rozsáhlého seznamu kategorií, byl tento krok velmi snadný.

Druhým a nejdůležitějším krokem bylo správné nastavení pravidel pro vyhledávání kolizí. Solibri Model Checker nabízí velké množství přednastavených kolizních pravidel. Ty bylo však potřeba doplnit o několik dalších. Mezi ty patřil například požadavek na minimální světlou výšku místnosti se sádkartonovým podhledem nebo na absenci definovaných prostupů ocelových nosníků stěnami.

Dalším fází již bylo vyhledávání kolizí programem. V tomto kroku byl zaznamenán reálný čas, po který software pracoval. Při řešení projektu X činil tento čas 4,15 sekundy, v případě projektu Y pak 17,26 sekundy. Tyto časy byly následně použity ve výpočtu.

Následně bylo nutné ověřit správnost vyhledaných kolizí. Vzhledem k tomu, že pravidla byla vytvořena převážně z již přednastavených sad, bylo nutné vyhledat jejich duplicity. Zároveň bylo nutné určit zda, se jedná či nejedná o kolizi, která byla vyžadována i od řešitelů. Software označoval i konflikty, jako je chybějící označení místností, nebo chybějící definice požárních vlastností. Těm sice většinou automaticky přiřadil nízkou závažnost, nicméně bylo nutné je identifikovat a nezahrnout je do výsledného počtu. Doba, kterou autor práce strávil kontrolou výsledků programu, byla zaznamenána a použita v následném výpočtu. Pro projekt X činila 20 minut, v případě projektu Y pak 38 minut.

Při sčítání celkového počtu kolizí bylo třeba očíslovat některé, programem agregované konflikty. Software kolize často slučoval do balíčků kolizí stejných vlastností zejména z důvodu přehlednosti výsledků. Doba, po kterou byla tato činnost vykonávána, byla zahrnuta do výše uvedených časů.

Celkový počet nalezených kolizí v projektu X činil 45 a v projektu Y 127. Jak již bylo řečeno, byly tyto počty uvažovány za konečné. Následně byl exportován report společně s prezentací s popisem a zobrazením jednotlivých kolizí. Tyto výstupy jsou obsaženy v přílohách této práce.



## 4 Srovnání zpracování úloh

V rámci vyhodnocení praktických úloh byl sestaven postup pro porovnání výkonu studentů a softwaru při řešení detekce kolizí. Bylo sestaveno několik rovnic, které vychází z principů statistických a fyzikálních veličin. V rámci výzkumu byla uvažována časová náročnost přípravy programu a pravidel pro vyhledávání, stejně tak jako časová náročnost na uvedení řešitelů do problematiky a vysvětlení zadání. Vztah mezi těmito hodnotami byl v této práci stanoven jako *Koeficient přípravy*. Ten je vyjádřen jako poměr obou hodnot, viz rovnice (4-1).

$$T = \frac{T_P}{T_S} \quad (4-1)$$

, kde  $T$  je koeficient přípravy,  $T_P$  je čas, potřebný k uvedení řešitelů do problematiky a vysvětlení zadání úloh a  $T_S$  je čas, potřebný ke správnému nastavení programu a pravidel pro vyhledávání kolizí

Pro účely této práce byly při výpočtu uvažovány shodné hodnoty u obou časů. Koeficient přípravy se tak rovnal jedné a neměl vliv na výsledek, viz rovnice (4-2). Důvodem byla reálná srovnatelnost obou časů u modelových úloh. Tento koeficient byl však obecně zaveden, aby bylo možné ve výpočtu reagovat na nepřiměřenou dobu přípravy řešitelů, respektive programu.

$$T_P = T_S \rightarrow T = \frac{T_S}{T_S} = \frac{1}{1} = 1 \quad (4-2)$$

, kde  $T$  je koeficient přípravy,  $T_P$  je čas, potřebný k uvedení řešitelů do problematiky a vysvětlení zadání úloh a  $T_S$  je čas, potřebný ke správnému nastavení programu a pravidel pro vyhledávání kolizí

Koeficient přípravy však lze použít i pro dosažení údajů pro dva rozdílné programy a přímo tak porovnat jejich nabídku nastavení, snadnost manipulace a další vlastnosti spojené s přípravou jejich práce.





## 4.1 Relativní účinnost detekce

Účinnost je fyzikální veličinou, běžně vyjadřovanou v procentech a definovanou jako poměr mezi výkonem a příkonem stroje, který vykonává práci. [25] Pro účely porovnání dvou odlišných subjektů, vyhledávajících kolize ve stavebních projektech, byla v této práci zavedena veličina *Relativní účinnost detekce*. Jedná se o poměr účinností dvou systémů, u kterých se uvažuje shodný příkon. Dá se tedy vyjádřit, jako poměr samotných výkonů dvou systémů viz rovnice (4-3). V případě této práce se jedná o výkon řešitelů a výkon softwaru při vyhledávání kolizí. Podíl je však upraven *koeficientem chybovosti*, zohledňujícím fakt, že řešitelé nemusí vždy nalézt všechny kolize, zatímco program ano.

$$\eta_r = \frac{\bar{P}}{\bar{S}} * C * T \quad (4-3)$$

, kde  $\eta_r$  je relativní účinnost detekce,  $\bar{P}$  je celkový průměrný výkon řešitelů a  $\bar{S}$  je průměrný výkon softwaru,  $C$  je koeficient chybovosti a  $T$  je koeficient přípravy

### 4.1.1 Koeficient chybovosti

Koeficient chybovosti představuje poměr mezi průměrným počtem kolizí nalezených řešiteli a celkovým počtem kolizí v modelu viz rovnice (4-4). Předpokládá se, že software určí všechny konflikty. Návrh řešení chybných výsledků programu je uveden v další kapitole (kapitola 4.1.3). Zároveň lze tento koeficient použít při případném srovnávání činnosti dvou programů. V případě, že celkový počet kolizí není znám, mohl by být přidán koeficient, upravující průměrný počet kolizí, nalezených softwarem.

$$C = \frac{C_p}{C_s} \quad (4-4)$$

, kde  $C$  je koeficient chybovosti,  $C_p$  je průměrný počet kolizí nalezených řešiteli a  $C_s$  je průměrný počet kolizí nalezených softwarem



Průměrný počet kolizí, nalezených řešiteli byl stanoven jako průměrná hodnota z počtů konfliktů, vyhledaných řešiteli na různých projektech v úlohách Q i A, viz rovnice (4-5).

$$C_P = \frac{\sum_{i=1}^n (\bar{Q}_i + \bar{A}_i)}{n} \quad (4-5)$$

, kde  $C_P$  je průměrný počet kolizí nalezených řešiteli,  $\bar{Q}_i$  je průměrný počet kolizí vyhledaných řešiteli na projektu  $i$  v úloze Q,  $\bar{A}_i$  je průměrný počet kolizí vyhledaných řešiteli na projektu  $i$  v úloze A,  $i$  nabývá hodnot dle zkoumaných projektů a  $n$  je počet variant dle řešených projektů a zároveň úloh.

Průměrný počet kolizí vyhledaných softwarem byl určen jako průměr z počtů kolizí vyhledaných softwarem na projektech, viz rovnice (4-6). V této úloze uvažujeme tento počet kolizí za celkový počet konfliktů v modelu.

$$C_S = \frac{\sum_{i=1}^n C_i}{n} \quad (4-6)$$

, kde  $C_S$  je průměrný počet kolizí nalezených softwarem,  $C_i$  je průměrný počet kolizí vyhledaných softwarem na projektu  $i$ ,  $i$  nabývá hodnot dle zkoumaných projektů a  $n$  je počet variant dle řešených projektů

#### 4.1.2 Určení průměrného výkonu řešitelů

Pro sestavení průměrného výkonu řešitelů bylo zapotřebí sloučit průměrné výkony vyhledávání kolizí na řešených projektech dle rovnice (4-7):

$$\bar{P} = \frac{\sum_{i=1}^n \bar{P}_i}{n} \quad (4-7)$$

, kde  $\bar{P}$  je průměrný výkon řešitelů,  $\bar{P}_i$  je průměrný výkon řešitelů na projektu  $i$ ,  $i$  nabývá hodnot dle zkoumaných projektů a  $n$  je počet variant dle řešených projektů



Jednotlivé výkony byly určeny pomocí zprůměrování výkonů na úlohách Q a A v rámci obou projektů, viz obecně vyjádřená rovnice (4-8).

$$\bar{P}_i = \frac{\bar{P}_i^Q + \bar{P}_i^A}{n} \quad (4-8)$$

, kde  $\bar{P}_i$  je průměrný výkon řešitelů na projektu  $i$ ,  $\bar{P}_i^Q$  je průměrný výkon řešitelů na projektu  $i$  v úloze Q a  $\bar{P}_i^A$  je průměrný výkon řešitelů na projektu  $i$  v úloze A,  $i$  nabývá hodnot dle zkoumaných projektů a  $n$  je počet variant dle řešených projektů

Fyzikální definice udává výkon jako veličinu, určující podíl určité práce za dobu, za kterou byla tato práce vykonána. [25] V této úloze byl výkon řešitelů definován jako podíl průměrného počtu nalezených kolizí za stanovený čas. Ten byl určen zvlášť pro každou z úloh Q a A. Zároveň byly stanoveny výkony dle projektů v úlohách Q a A viz rovnice (4-9) a (4-10). Pokud není uvedeno jinak, časové údaje jsou vždy uvažovány v jednotkách sekund.

$$\bar{P}_i^Q = \frac{\bar{Q}_i}{t_Q} \quad (4-9)$$

, kde  $\bar{P}_i^Q$  je průměrný výkon řešitelů na projektu  $i$  v úloze Q,  $\bar{Q}_i$  je průměrný počet nalezených kolizí v úloze Q na projektu  $i$ ,  $i$  nabývá hodnot dle zkoumaných projektů a  $t_Q$  je čas pro vypracování úlohy Q

$$\bar{P}_i^A = \frac{\bar{A}_i}{t_A} \quad (4-10)$$

, kde  $\bar{P}_i^A$  je průměrný výkon řešitelů na projektu  $i$  v úloze A,  $\bar{A}_i$  je průměrný počet nalezených kolizí v úloze A na projektu  $i$ ,  $i$  nabývá hodnot dle zkoumaných projektů a  $t_A$  je čas pro vypracování úlohy A



Pro výpočet průměrných výkonů bylo třeba vypočítat průměrný počet vyhledaných kolizí na jednoho řešitele. Ten je definován jako podíl součtu vyhledaných kolizí a počtu řešitelů. Bylo zapotřebí stanovit tyto hodnoty po projektech v úlohách Q a A, viz rovnice (4-11) a (4-12).

$$\bar{Q}_i = \frac{\sum_{i=1}^n Q_i}{p_i} \quad (4-11)$$

, kde  $\bar{Q}_i$  je průměrný počet nalezených kolizí v úloze Q na projektu  $i$ ,  $\sum Q_i$  je součet vyhledaných kolizí v úloze Q na projektu  $i$ ,  $i$  nabývá hodnot dle zkoumaných,  $n$  je počet variant dle řešených projektů a  $p_i$  je počet řešitelů projektu  $i$

$$\bar{A}_i = \frac{\sum_{i=1}^n A_i}{p_i} \quad (4-12)$$

, kde  $\bar{A}_i$  je průměrný počet nalezených kolizí v úloze A na projektu  $i$ ,  $\sum A_i$  je součet vyhledaných kolizí v úloze A na projektu  $i$ ,  $i$  nabývá hodnot dle zkoumaných,  $n$  je počet variant dle řešených projektů a  $p_i$  je počet řešitelů projektu  $i$

#### 4.1.3 Určení průměrného výkonu softwaru

Pro sestavení průměrného výkonu softwaru bylo zapotřebí sloučit a zprůměrovat výkony vyhledávání kolizí na projektech X a Y viz rovnice (4-13).

$$\bar{S} = \frac{\sum_{i=1}^n S_i}{m} \quad (4-13)$$

, kde  $\bar{S}$  je průměrný výkon softwaru,  $S_i$  je výkon softwaru na projektu  $i$ ,  $i$  nabývá hodnot dle zkoumaných,  $n$  je počet variant dle řešených projektů a  $m$  je počet variant dle řešených projektů

Jednotlivé výkony byly stanoveny jako podíly počtu detekovaných kolizí a času, po který software kolize vyhledával. Ten byl však oproti předchozímu postupu doplněn o *revizní dobu*. Toto prodloužení práce programu je uvažováno z důvodu nutnosti kontroly výsledků. Uživatel totiž dostává velké množství vyhledaných konfliktů, z nichž některé nemusí být ve skutečnosti kolizemi. Je tedy zapotřebí výsledky kontrolovat, na což je obecně potřeba určitého času a práce. S ohledem na možnost použití rovnic i pro jiné programy, jeví se přidání revizní doby jako praktická úprava. Reflektuje totiž modifikaci výkonu programu dle jeho schopností, nedokonalého nastavení pravidel pro



vyhledávání kolizí, atp. V praxi reprezentuje revizní doba čas v minutách, který byl vynaložen pro kontrolu výsledků softwaru.

Výkon programu při vyhledávání kolizí na projektu byl stanoven dle vzorce (4-14):

$$S_i = \frac{C_i}{t_i + 60 * t_R^i} \quad (4-14)$$

, kde  $S_i$  je výkon softwaru na projektu  $i$ ,  $C_i$  je počet softwarem vyhledaných kolizí v projektu  $i$ ,  $t_i$  je čas, po který program vyhledával kolize v projektu  $i$  a  $t_R^i$  je revizní doba v minutách, potřebná pro kontrolu výsledků na projektu  $i$  a  $i$  nabývá hodnot dle řešených projektů

Pomocí těchto rovnic je tedy možné určit výkon určitého programu při vyhledávání kolizí. Předpokladem je však fakt, že software vyhledá všechny kolize. Při zavádění koeficientu efektivity bylo uvažováno s možností, kdy program nalezne nejen všechny skutečné kolize, ale označí i konflikty, které jsou následně vyhodnoceny jako správně řešené. Varianta, kdy software vlivem špatného nastavení pravidel nenalezne všechny správné konflikty, by mohla být zahrnuta opět v rámci zavedení koeficientu. Je však otázkou, do jaké míry je toto chyba či nedostatek softwaru. Jako možnost k navázání této práce tak je výzkum zavedení dalšího koeficientu chybovosti.

## 4.2 Číselné vyhodnocení relativní účinnosti

V této kapitole je popsán výpočet relativní účinnosti detekce dle modelových úloh sestavených pro tuto diplomovou práci. Označení všech hodnot v této kapitole se shoduje s jejich označením a popisem v kapitole předchozí (kapitola 4.1). Vstupní hodnoty byly shromážděny dle postupu, uvedeném v dřívější kapitole (kapitola 3). Ty byly sepsány do tabulky (tabulka 2).

Počty kolizí vyhledaných řešiteli v úlohách  $A_x$ , respektive  $A_y$ , byly určeny součtem obou úloh  $A$  i  $Q$ . To vyplývá ze zadání těchto úloh, kdy měli řešitelé za úkol označit kolize nejprve v úloze na rychlost (úloha  $Q$ ) a poté již ty samé kolize neoznačovat při zpracovávání úlohy na přesnost (úloha  $A$ ). V tabulce přehledu výsledků byl počet kolizí vyhledaných pouze v samotné úloze  $A$  označen písmenem  $a$ . Součtem s počty konfliktů, nalezených v úloze  $Q$  je pak hledaný údaj, označený jako  $A_x$ , respektive  $A_y$ .



Tabulka 2: Vstupní hodnoty pro výpočet relativní účinnosti detekce

Zdroj: vlastní tvorba

Vstupní hodnoty			
ŘEŠITELÉ		SOFTWARE	
PROJEKT X			
Qx	131 kolizí	Cx	45 kolizí
Ax	213 kolizí	tx	4,15 sekund
nx	21 lidí		
PROJEKT Y			
Qy	133 kolizí	Cy	127 kolizí
Ay	284 kolizí	ty	17,26 sekund
ny	22 lidí		
ČASY			
tq	120 sekund	Trx	20 minut
ta	420 sekund	Try	38 minut

Postupným dosazováním hodnot do rovnic (4-7) až (4-12) byly stanoveny výsledky pro zjištění průměrného výkonu řešitelů. Jejich přehled byl sepsán do tabulky, viz níže (Tabulka 3).

Tabulka 3: Výsledky výpočtu průměrného výkonu řešitelů

Zdroj: vlastní tvorba

Výpočet průměrného výkonu řešitelů ( $\bar{P}$ )			
číslo rovnice	hodnota	výsledek	jednotka
(4-12)	$\bar{A}_x$	<b>10,143</b>	kolizí
	$\bar{A}_y$	<b>12,909</b>	kolizí
(4-11)	$\bar{Q}_x$	<b>6,238</b>	kolizí
	$\bar{Q}_y$	<b>6,045</b>	kolizí
(4-10)	$\bar{P}_x^A$	<b>0,0241</b>	kolize/s
	$\bar{P}_y^A$	<b>0,0307</b>	kolize/s
(4-9)	$\bar{P}_x^Q$	<b>0,0520</b>	kolize/s
	$\bar{P}_y^Q$	<b>0,0504</b>	kolize/s
(4-8)	$\bar{P}_x$	<b>0,0381</b>	kolize/s
	$\bar{P}_y$	<b>0,0406</b>	kolize/s
(4-7)	$\bar{P}$	<b>0,0393</b>	kolize/s

Dosazováním vstupních hodnot z předchozí tabulky (tabulka 2) byly určeny výsledky, potřebné k výpočtu průměrného výkonu softwaru. Hodnoty, vyplývající z příslušných rovnic (4-13) a (4-14) byly sepsány do tabulky (tabulka 4).

Tabulka 4: Výsledky výpočtu průměrného výkonu softwaru

Zdroj: vlastní tvorba

Výpočet průměrného výkonu softwaru ( $\bar{S}$ )			
číslo rovnice	hodnota	výsledek	jednotka
(4-14)	Sx	<b>0,0374</b>	kolize/s
	Sy	<b>0,0553</b>	kolize/s
(4-13)	$\bar{S}$	<b>0,0463</b>	kolize/s

Po dosazení hodnot z předchozích tabulek (tabulka 2, tabulka 3 a tabulka 4) do příslušných rovnic ((4-2) až (4-6)), byly zjištěny výsledky výpočtu relativní účinnosti detekce. Ty byly sepsány do tabulky (tabulka 5), viz níže.

Tabulka 5: Výsledky výpočtu relativní účinnosti detekce

Zdroj: vlastní tvorba

Výpočet relativní účinnosti detekce ( $\eta_r$ )			
číslo rovnice	hodnota	výsledek	jednotka
(4-6)	Cs	<b>86,000</b>	kolizí
(4-5)	Cp	<b>8,834</b>	kolizí
(4-4)	C	<b>0,103</b>	-
(4-2)	T	<b>1</b>	-
(4-3)	$\eta_r$	<b>8,72%</b>	-

Výsledkem výpočtu je relativní účinnost detekce kolizí řešitelů vůči činnosti BIM nástroje. Její hodnota činí 8,72%. Z toho vyplývá, že řešitelé v úlohách dosáhli necelé desetiny výkonnosti programu Solibri Model Checker. Lze očekávat pokles této efektivity při řešení komplexnějších a složitějších projektů. I přes delší dobu, která je potřebná ke kontrole výsledků BIM softwaru, jeví se jeho použití mnohem efektivnější, než klasický postup při vyhledávání konfliktů v návrhu projektu. Podrobnější shrnutí výsledků uvedeno v závěru této práce.



## ZÁVĚR

V úvodu této diplomové práce byla popsána problematika práce s BIM modely, její základní principy a přínosy (Cíl 1). Ty mají podobu převážně finančních a časových úspor. Jejich generování souvisí se řešením změn projektu už ve fázi návrhu, čili jsou realizovány podstatně dříve, než při klasickém vedení projektu.

Dále byly uvedeny současné trendy, týkající se BIM ve stavebním průmyslu po celém světě. Byly popsány případy využití BIM v různých státech, jejichž přístupy by se mohla Česká republika inspirovat při zavádění metodiky jakožto součásti veřejných zakázek. To je považováno za nezbytné pro celkovou implementaci BIM do výstavbových procesů na tuzemském trhu. Na základě usnesení vlády o významu metody BIM pro stavební praxi lze reálně očekávat postupný nárůst podpory ze strany státu.

Následující část úvodní kapitoly byla věnována procesu kontroly informačních modelů. Byl zdůrazněn fakt, že se i přes značnou automatizaci postupů nelze obejít bez zásahu člověka. Ten spočívá ve finální kontrole výstupů a dohledu nad činností BIM softwarů. Zároveň není možné nasimulovat selhání pracovníků na staveništi a je tak potřeba některé kolize řešit operativně in-situ. Dále byla uvedena koncepce kontroly modelu se zavedením názvu Revize modelu. Její obsah byl rozdělen na čtyři části, z nichž každá poskytuje různé náměty k vývoji procesu analýzy a tvorby BIM modelů. Koncepce Revize modelu není v České republice stanovena a nabízí tak prostor pro zavádění nových pojmů a metodik do procesu kontrolních činností a tvorby BIM modelů ve výstavbových projektech.

V další části byl detailně definován kolize v informačním modelu, uvedeny historické souvislosti a sepsáno typové rozdělení kolizí na tvrdé, měkké, 4D a kolize animovaných objektů. Každá z těchto variant byla podrobně popsána. Současně byly uvedeny konkrétní příklady kolizí včetně jejich grafického znázornění.

Dále byly popsány možné dopady kolizí při jejich zanedbání. Z uvedených informací z případových studií zahraničních stavebních společností vyplývá, že náklady na dodatečné napravování chyb v návrhu jsou několikrát vyšší než ty, které společnost vynaloží na zajištění procesu detekce kolizí. Zdroje hovoří až o 300% ROI spojené s používáním detekce kolizí. V rámci této kapitoly byl uveden i možný dopad kolizí na bezpečnost pracovníků. Byly uvedeny praktické příklady využití BIM nástrojů pro celkové zlepšení BOZP, například pomocí propojení BIM modelu a GPS přijímačů na vestách pracovníků.

Následně byly popsány různé přístupy řešení kolizí. Důrazem byl kladen na stálou nutnost lidského dozoru. Jako příklad byla uvedena kolize vedení vzduchotechniky a střešní konstrukce na rekonstrukci rodinného domu. Autor se přímo podílel na řešení tohoto konfliktu a jím zpracovaný postup je v kapitole uveden.





V druhé části diplomové práce byl proveden průzkum programů na detekci kolizí pomocí BIM (Cíl 2). Byla stanovena kritéria pro výběr vhodných softwarů. Ta se týkala především kompatibility s datovým formátem IFC z důvodu jeho otevřenosti a samostatné formy programu – tzv. *stand-alone*. Účelem bylo vybrat specializované, schopné nástroje, které by bylo možné doporučit případným zájemcům o detekci kolizí pomocí BIM. Poté byla sestavena přehledová tabulka s vybranými programy. Těmi byly: *Navisworks Manage*, *Tekla BIMsight*, *Solibri Model Checker*, *BIMReview* a *Bentley Navigator Connect*. Zároveň byly v tabulce stručně okomentovány vlastnosti, relevantní k tématu práce. Výběr právě těchto programů byl ovlivněn mimo jiné i jejich dostupností. Z různých důvodů nebylo možné vyzkoušet další nástroje, které dle jejich vývojářů splňovaly požadovaná kritéria. Konkrétně se jednalo o *Synchro Pro*, *DDS CAD Viewer* a *IFC Viewer*. V tomto ohledu nabízí práce prostor k navázání v podobě další rešerše nástrojů a jejich přezkoumání.

Dále byla provedena detailní analýza možností a schopností vybraných programů (Cíl 3) s výjimkou *Bentley Navigator Connect*, který byl, i přes snahu autora, kvůli technickým problémům s importováním IFC souborů vynechán. Průzkum zbylých nástrojů byl zaměřen na funkce přímo spojené s detekcí kolizí, jejich označováním, programováním pravidel pro jejich vyhledávání, komunikací s delegovanými spolupracovníky a možnost exportu informací. Jako nejvhodnější nástroje byly označeny *Navisworks Manage* a *Solibri Model Checker*. Ten však výrazně vyčníval zpracováním programování pravidel kolizí a oproti *Navisworks Manage* se jevil více specializovaným. Na základě těchto poznatků, pozitivních ohlasů z praxe a subjektivního dojmu byl *Solibri Model Checker* vybrán pro použití do procesu praktického srovnávání efektivity detekce kolizí.

V další kapitole této práce byl navržen praktický postup, kterým lze porovnat činnosti dvou různých řešitelů stejné úlohy na detekci kolizí. V tomto případě byl záměr porovnat efektivitu lidských řešitelů oproti specializovanému softwaru a zjistit tak případné nedostatky. Nejprve byly autorem zpracovány modely dvou projektů, záměrně obsahujících různé počty konfliktů. Následně byly exportovány výkresy jako podklad pro lidské řešitele a IFC soubory pro import do *Solibri Model Checker*. Bylo nutné vybrat dostatečně reprezentativní vzorek populace. Vzhledem k problémům s obstaráváním vzorku profesionálů byli pro své znalosti vybráni studenti prvního ročníku magisterského studia v oboru Projektový management a inženýring Fakulty Stavební, ČVUT. Těm byly zadány dvě úlohy na rychlost a přesnost vyhledávání kolizí pro oba projekty a zároveň jim bylo autorem vysvětleno zadání pomocí krátké prezentace. Studenti následně zpracovali zadané úkoly a jejich výsledky byly použity ve výpočtu. Následně byly stejné projekty zpracovány pomocí *Solibri Model Checker*. Společně s časem vynaloženým na kontrolu představovaly výsledky programu zbylé vstupní hodnoty výpočtu.



Poslední část této práce je věnována autorem sestavenému výpočtu, který je tvořen několika rovnicemi. V rámci výpočtu byla stanovena veličina relativní účinnosti detekce kolizí, která udává procentuální efektivitu jednoho řešitele oproti druhému. Dále byly zavedeny koeficienty přípravy a chybovosti, které zohledňují čas na přípravu, respektive účinnost vyhledávání konfliktů. Dosazením vstupních hodnot z praktických úloh do rovnic byla vypočtena hodnota relativní účinnosti detekce řešitelů ve výši 8,72%. To znamená, že software je více než desetinásobně efektivnější, než průměr vzorku lidských řešitelů. I přes nárůst doby, potřebné ke kontrole výsledků programu by jeho účinnost stoupala, neboť se dá předpokládat, že by lidský řešitel více chyboval. Zároveň lze očekávat, že účinnost detekce řešitele bude klesat se stoupající komplexností projektu. Negativní vliv také může mít například nekvalitní projektová dokumentace profesí, absence sjednocení úrovně detailů napříč profesemi, atp. Účinnost detekce programem může být značně ovlivněna použitím rozdílných knihoven objektů projektanty. Obecně lze říct, že v obou případech má velký vliv celková kvalita projektových dokumentací, jednotné postupy a principy a kvalitní komunikace mezi zúčastněnými.

Cílem této práce bylo vyhodnocení přínosu použití BIM nástrojů při detekci kolizí (Cíl 4). Na základě provedeného srovnání a podstatnému významu kolizí lze říci, že je tento přínos značný. Je otázkou, zda se například menší společnosti vyplatí investovat nemalou částku do podobného softwaru při relativně nízkých hodnotách projektů. Díky širokému výběru je však v dnešní době možné zvolit i levnější nástroje, ba dokonce i ty, které jsou k dispozici zdarma. Z průzkumu v této práci je sice nemůžeme označit za srovnatelné se špičkou, ale určitému typu uživatelů mohou jejich funkce stačit.

Na tuto práci lze navázat ve více rovinách. Jak již bylo zmíněno, určitá část programů na detekci kolizí nemohla být z různých důvodů zahrnuta do analýz, a proto je možné v dalších pracích zkoumat právě je. Dále se nabízí návaznost z hlediska rozvoje koncepce Revize modelu. Vzhledem k tomu, že byly v modelových úlohách uvažovány pouze určité typy kolizí (tvrdé a měkké), je možné navázat podobným procesem, který by zahrnoval zkoumání 4D kolizí s propojením na harmonogram výstavby a kolizí animovaných objektů. Lze předpokládat nárůst dominance efektivity softwaru, neboť zmíněné kolize jsou ručně složitě zpracovatelné a hůře se předvídají. Další možností návaznosti může být výzkum hranice, kdy se společnosti vyplatí investovat do metodiky BIM a převážně pak do detekce kolizí. To vždy závisí na velikosti jak projektu, tak společnosti. Je možné také rozvést uvedené rovnice tak, aby zohledňovaly přesněji některé aspekty, jako například čas na přípravu řešitelů či jejich úpravu na přesnější porovnání dvou softwarů



## Reference

1. ČERNÝ, M., et al. *BIM příručka* [Elektronická kniha]. Praha: Odborná rada pro BIM a.s., 2013. ISBN 978-80-260-5279-5. Dostupné také z: <http://issuu.com/czbim/docs/bim-prirucka-2013-v1>
2. THE BUILDING AND CONSTRUCTION AUTHORITY - MINISTRY OF NATIONAL DEVELOPMENT OF SINGAPORE. CORENET e-Submission System. *CORENET- Construction and Real Estate Network* [online]., 2016 [cit. 2016-11-14]. Dostupné z: <https://www.corenet.gov.sg/general/corenet-e-submission-system.aspx>
3. MATĚJKA, P. *Základy informačního modelování* [prezentace]. Praha: Fakulta Stavební ČVUT v Praze, 2015 [cit. 2015-04-28].
4. ÚŘAD VLÁDY ČESKÉ REPUBLIKY. Informační systém ODok. [Usnesení č. 958 - 1353/16 Význam metody BIM (Building Information Modelling) pro stavební praxi v ...] In: *Jednání vlády* [online]. 2. 11. 2016 [cit. 2016-10-05]. Dostupné z: <https://apps.odok.cz/djv-agenda?date=2016-11-02>
5. HJELSETH, E. a NISBET, N. Overview of concepts for model checking. Káhira, Egypt: Norwegian University of Life Sciences, 2010.
6. OBI, I. What is clash detection? How does BIM help? In: *The BIM Center* [online]. 2016 [cit. 2016-08-20]. Dostupné z: <http://www.thebimcenter.com/2016/03/what-is-clash-detection-how-does-bim-help.html>
7. HI-TECH CADD SERVICES. 3 Types Of 3D BIM Clash Detection Have Their Own Importance. *Hi-Tech CADD Services* [online]., 2014, verze 9.10.2014 [cit. 2016-11-02]. Dostupné z: <http://www.hitechcaddservices.com/news/3-types-of-3d-clash-detection-have-their-own-importance/>
8. REDDY, P. *BIM for building owners and developers: Making a business case for using BIM on projects*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2012. ISBN 978-0-470-90598-2.
9. KYMMELL, W. *Building Information Modelling: Planning and Managing Construction Projects with 4D CAD and ....* New York: The McGraw-Hill Companies, Inc., 2008. ISBN 0-07-149453-7.
10. HI-TECH CADD SERVICES. BIM for Clash Detection and Risk Analysis Streamlines the Construction Process. *Hi-Tech CADD Services* [online]., 2014, verze 14.11.2014 [cit. 2016-11-02]. Dostupné z: <http://www.hitechcaddservices.com/news/bim-for-clash-detection-and-risk-analysis-streamlines-the-construction-process/>
11. THE WASTE AND RESOURCES ACTION PROGRAMME. Case study: Achieving resource efficiency through BIM. In: *The Waste and Resources Action Programme* [online]. [cit. 2016-12-01]. Dostupné z: [http://www.wrap.org.uk/sites/files/wrap/Leeds\\_Arena\\_FINAL.pdf](http://www.wrap.org.uk/sites/files/wrap/Leeds_Arena_FINAL.pdf)
12. LEFEVRE, M. Build Smarter, Faster, and Cheaper with BIM. In: *Buildings: Smarter Facility Management* [online]. 07. 01. 2008 [cit. 2016-12-05]. Dostupné z: <http://www.buildings.com/article-details/articleid/6149/title/build-smarter-faster-and-cheaper-with-bim>
13. TESSERACT-DESIGN. Web Tesseract Design. *BIM Coordination During Design: Do It Right the First Time – Using BIM to Coordinate Systems During ...* [online]., 2014 [cit. 2016-12-06]. Dostupné z: <http://www.tesseract-design.com/bim-coordination-during-design.html>
14. REDPOINT POSITIONING CORPORATION. Manage Construction Operations in Real Time. *Redpoint Positioning* [online]., 2015, verze 21.9.2015 [cit. 2016-11-14]. Dostupné z: <https://www.redpointpositioning.com/rtils-for-vertical-industries/rtils-for-construction/>



15. EASTMAN, C., TEICHOLZ, P., SACKS, R. a LISTON, K. *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers, ...* second edition. Hoboken, New Jersey: John Wiley & Sons, 2011. ISBN 978-0-470-54137-1.
16. ŠABART, J. *Stavební úpravy rodinného domu: dokumentace pro provedení stavby* [výkres]. Praha: 2016.
17. AUTODESK, INC. Navisworks: Subscribe. *Autodesk Subscribe* [online]., 2016, verze 24.11.2016 [cit. 2016-11-24]. Dostupné z: <http://www.autodesk.com/products/navisworks/subscribe>
18. TRIMBLE SOLUTIONS CORPORATION. Download Tekla Bimsight. *Trimble* [online]., 2016, verze 24.11.2016 [cit. 2016-11-24]. Dostupné z: <http://www.teklabimsight.com/download>
19. SOLIBRI, INC. *Solibri Solution Center* [online]. Solibri, Inc., 2016, verze 24.11.2016 [cit. 2016-11-24]. Dostupné z: <https://solution.solibri.com/> - nutné přihlášení
20. WATSON, S. Interview. In: *Sales management*. StruMIS LTD, 5. prosince 2016, 10:37.
21. SÝKORA, P. Navigator CONNECT (nejen) česky. *CAD Lexikon* [online]., 2016, verze 12.1.2016 [cit. 2016-12-10]. Dostupné z: <http://www.cadlexikon.sk/cz/vsimnite-si-v-novej-verzii/connect/377-navigator-connect>
22. AUTODESK, INC. Autodesk Navisworks Help: Glossary. *Autodesk Knowledge Network* [online]., 2016, verze 26.11.2016 [cit. 2016-11-26]. Dostupné z: <https://knowledge.autodesk.com/support/navisworks-products/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Navisworks/files/GUID-27EA59E6-1A15-4372-9D7D-90508936B512-htm.html>
23. BENEŠ, J. BIM model pod rentgenem. In: *Archinews* [online]. 11. 2. 2014, verze 11.2.2014 [cit. 2016-09-20]. Dostupné z: <http://www.archinews.cz/10-202-krok-za-krokem-bim-model-pod-rentgenem.aspx#.WffHq1PhC00>
24. GRAPHISOFT SE. Partner Solutions: Solibri Model Checker. *Web Graphisoft SE*. [online]., 2016 [cit. 2016-08-16]. Dostupné z: [http://www.graphisoft.com/archicad/partner\\_solutions/solibri\\_model\\_checker/](http://www.graphisoft.com/archicad/partner_solutions/solibri_model_checker/)
25. REICHL, J. a VŠETIČKA, M. Výkon, příkon, účinnost. *Encyklopedie fyziky* [online]. [cit. 2016-12-10]. Dostupné z: <http://fyzika.jreichl.com/main.article/view/51-vykon-prikon-ucinnost>
26. TICHÝ, M. a ERBEN, P. *Ovládní rizika. Analýza a management*. Praha: C.H. Beck, 2006. ISBN 80-7179-415-5.



## Seznam tabulek

Tabulka 1: Přehled vybraných programů pro práci s kolizemi v BIM modelech .....	24
Tabulka 2: Vstupní hodnoty pro výpočet relativní účinnosti detekce .....	53
Tabulka 3: Výsledky výpočtu průměrného výkonu řešitelů .....	53
Tabulka 4: Výsledky výpočtu průměrného výkonu softwaru .....	54
Tabulka 5: Výsledky výpočtu relativní účinnosti detekce .....	54

## Seznam obrázků

Obrázek 1: Náklady na změny projektu .....	4
Obrázek 2: Schéma konceptu Revize modelu .....	6
Obrázek 3: Ukázka tvrdé kolize mezi potrubím a stěnou .....	11
Obrázek 4: Ukázka tvrdé kolize mezi schodištěm a vodorovnou konstrukcí .....	11
Obrázek 5: Ukázka měkké kolize nedostatečného uložení nosníku .....	13
Obrázek 6: Ukázka měkké kolize sloupu umístěného před prvek okna .....	13
Obrázek 7: Ukázka měkké kolize nedostatečného prostoru mezi klozetem a dveřmi... ..	14
Obrázek 8: Ukázka měkké kolize v nedostatečných rozměrech před nástupním a výstupním stupněm schodiště .....	14
Obrázek 9: Detail půdorysu podkrovního bytu se zaměřením na místo kolize vedení vzduchotechniky .....	18
Obrázek 10: Fotografie prostoru kolize vedení vzduchotechniky, pohled z kuchyně....	18
Obrázek 11: Fotografie prostoru kolize vedení vzduchotechniky, pohled z koupelny... ..	19
Obrázek 12: Názorné zobrazení variant vedení vzduchotechniky.....	19
Obrázek 13: Schématické znázornění rozsahu IFC vzhledem k ISO 16739.....	22
Obrázek 14: Ukázka nastavení atributů pravidel.....	27
Obrázek 15: Ukázka uživatelského rozhraní nastavení detekce kolizí v programu Navisworks Manage .....	28
Obrázek 16: Ukázka znázornění kolize programem Navisworks Manage .....	29
Obrázek 17: Zobrazení vyhledaných kolizí v rozhraní Navisworks Manage .....	30
Obrázek 18: Ukázka reportu kolizí z Navisworks Manage ve formátu HTML.....	30
Obrázek 19: Ukázka možností nastavení pravidel v Tekla BIMsight .....	31
Obrázek 20: Ukázka označování kolizí v programu Tekla BIMsight .....	32
Obrázek 21: Ukázka konverzačního kanálu v podání Tekla BIMsight .....	33
Obrázek 22: Ukázka programování pravidel v programu Solibri Model Checker .....	36
Obrázek 23: Ukázka vyhledané kolize v Solibri Model Checker .....	37
Obrázek 24: Ukázka popisu kolize pomocí Solibri Model Checker .....	38
Obrázek 25: Ukázka reportu pomocí Solibri Model Checker .....	39
Obrázek 26: Ukázka přehledu kolizí a jejich znázornění v modelu pomocí BIMReview	41



## Seznam rovnic

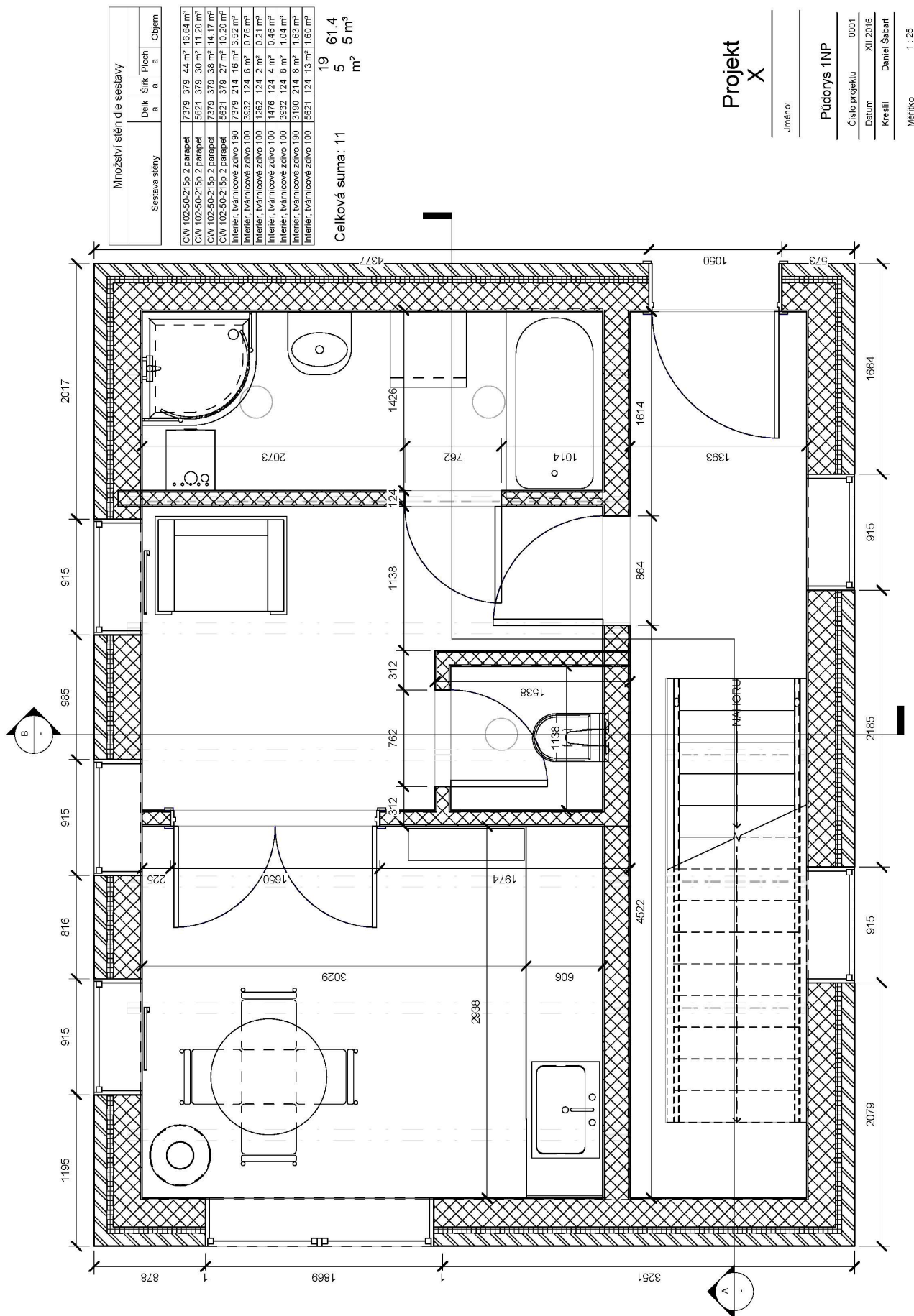
(4-1).....	47
(4-2).....	47
(4-3).....	48
(4-4).....	48
(4-5).....	49
(4-6).....	49
(4-7).....	49
(4-8).....	50
(4-9).....	50
(4-10).....	50
(4-11).....	51
(4-12).....	51
(4-13).....	51
(4-14).....	52

## Seznam příloh

1. Výkres projektu X – půdorys 1NP (i elektronicky)
2. Výkres projektu X – řezy A a B (i elektronicky)
3. Výkres projektu Y – půdorys 1NP (i elektronicky)
4. Výkres projektu Y – řezy A a B (i elektronicky)
5. Výsledky úloh řešitelů
6. Report projektu X – Solibri Model Checker (i elektronicky)
7. Report projektu Y – Solibri Model Checker (i elektronicky)
8. Model projektu X – *Revit Architecture* (elektronicky)
9. Model projektu Y – *Revit Architecture* (elektronicky)
10. Model architektonický projektu X – IFC (elektronicky)
11. Model statický projektu X – IFC (elektronicky)
12. Model TZB projektu X – IFC (elektronicky)
13. Model architektonický projektu Y – IFC (elektronicky)
14. Model statický projektu Y – IFC (elektronicky)
15. Model TZB projektu Y – IFC (elektronicky)
16. SMC model projektu X (elektronicky)
17. SMC model projektu Y (elektronicky)
18. Zadání diplomové práce (elektronicky)



# Příloha 1: Výkres projektu X – půdorys 1NP

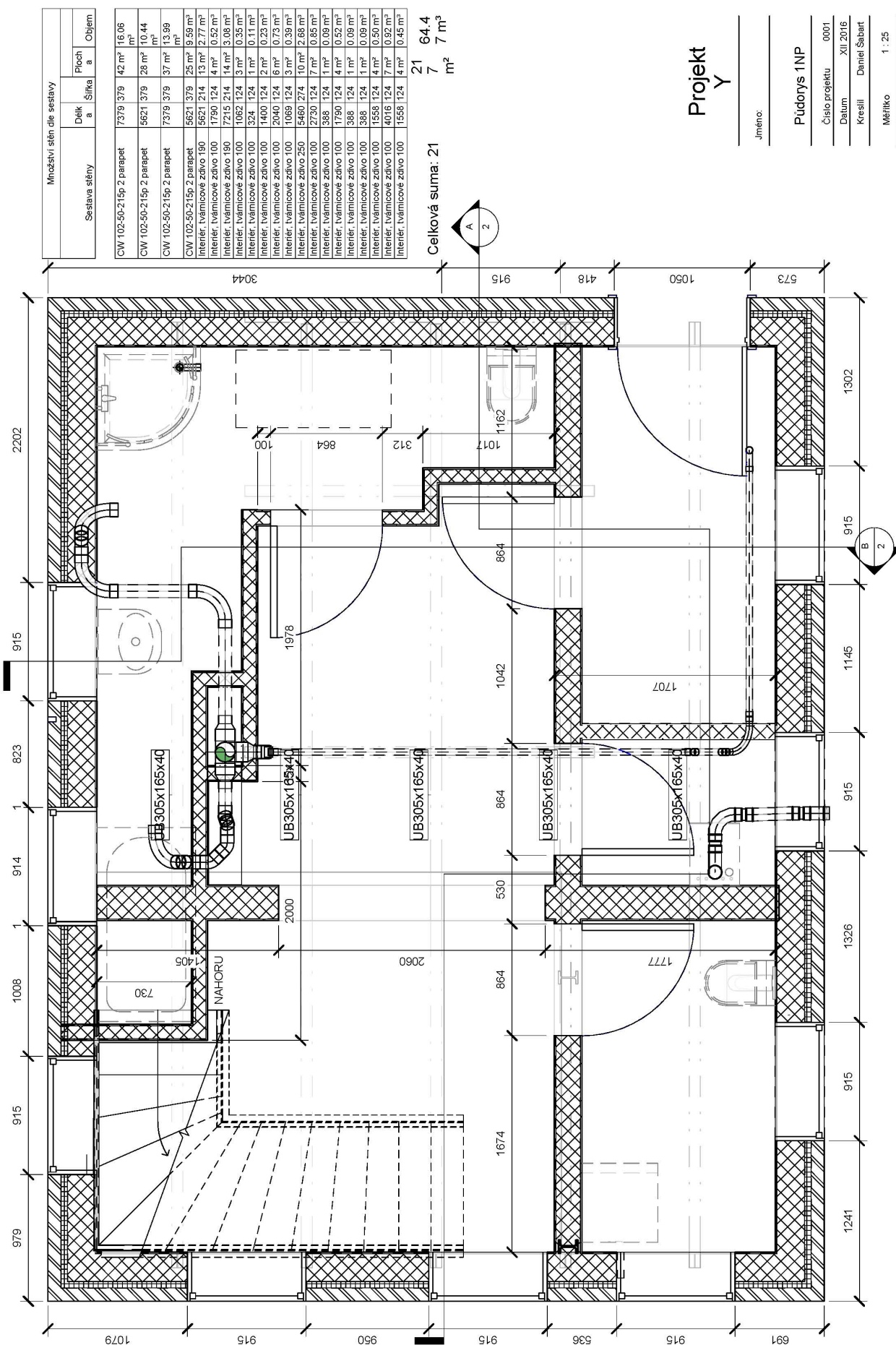






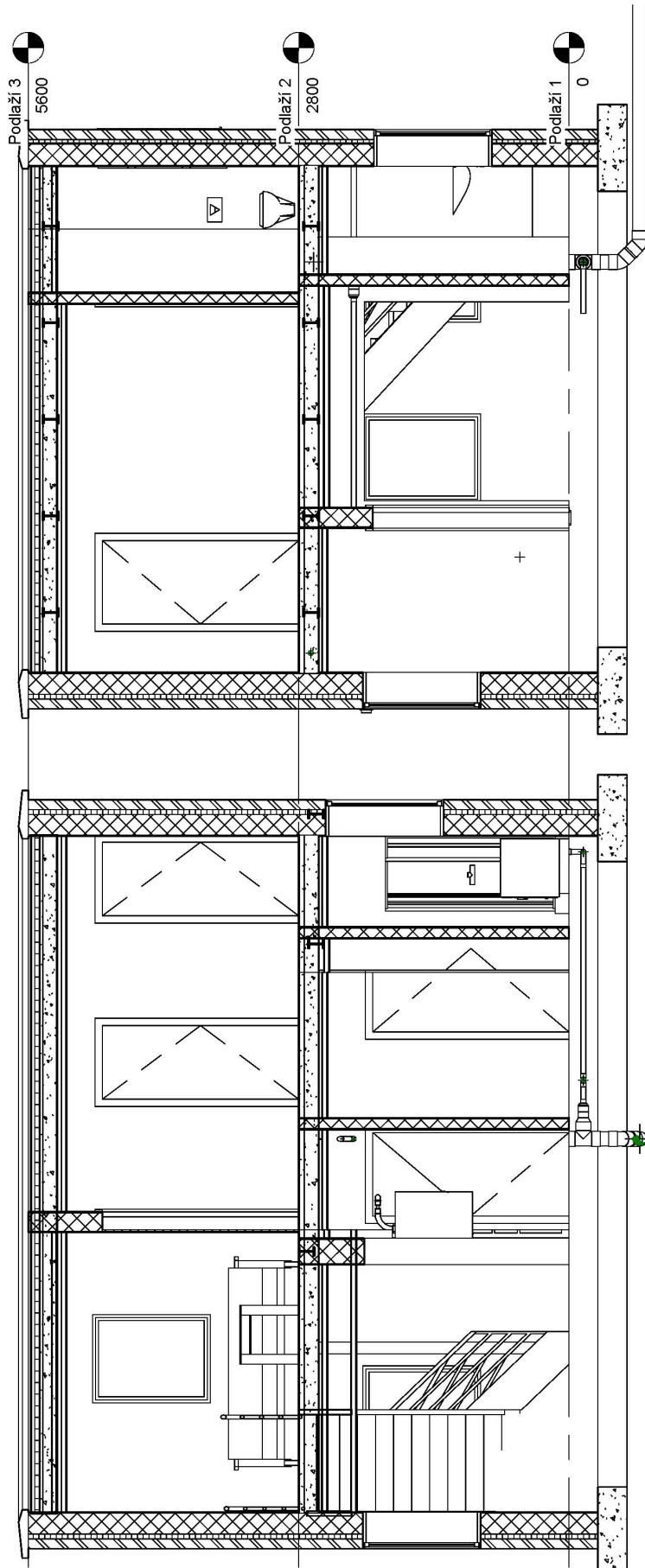


## Příloha 3: Výkres projektu Y – půdorys 1NP





## Příloha 4: Výkres projektu Y – řezy A a B



**Řez A**  
1 : 40

**Řez B**  
1 : 40

Množství konstrukčních nosníků a vzpěr		
Konstrukční prvek	Typ	Délka

Počty dveří						
Kód sestavy	Popis sestavy	Počet	Popis dveří	Typ	Sířka	Výška
		2	M_Jednoduché_hlaďké	0762 x 2032mm	762	2032
		5	M_Jednoduché_hlaďké	0864 x 2032mm	864	2032
B2030100	Glazed Doors & Entrances	1	Dveře_vnější_jednokřídelé_vhodové_poločlenní_odtok_sko_dřevo_částad	900 x 2000mm	1000	2000

**Celková suma**

Množství oken						
Počet	Popis okna	Typ	Sířka	Výška	Typ ohna	Komentáře
16	M_Pevné	0815 x 1220mm	915	1220	15	
1	M_Pevné	0815 x 1500mm	2	1500	26	
1	Okenní křídlo_čvojtité_s_listou	0812 x 1220mm	812	1220	24	

**Celková suma**

UB univerzální nosníky		
UB univerzální nosník	UB305x165x40	2000
UB univerzální nosník	UB305x165x40	7509
UB univerzální nosník	UB305x165x40	7550
UB univerzální nosník	UB305x165x40	7550
UB univerzální nosník	UB305x165x40	7509
UB univerzální nosník	UB305x165x40	7509
UB univerzální nosník	UB305x165x40	7550
UB univerzální nosník	UB305x165x40	2700
UB univerzální nosník	UB305x165x40	2700
UB univerzální nosník	UB305x165x40	7238
UB univerzální nosník	UB305x165x40	7238
UB univerzální nosník	UB305x165x40	7238
UB univerzální nosník	UB305x165x40	7238
UB univerzální nosník	UB305x165x40	97624

**UB305x165x40: 16**

**UB univerzální nosníky: 16**

**Celková suma: 16**

**Projekt Y**

Jméno:

**Řezy A a B**

Číslo projektu	0001
Datum	XII. 2016
Kresil	Daniel Šabart
Měřítko	1 : 40



## Příloha 5: Výsledky úloh řešitelů

Výsledky úloh						
Projekt	X			Y		
počet řešitelů	21			22		
Úloha	Q	a	A	Q	a	A
Počet nalezených kolizí	8	1	9	5	5	10
	8	3	11	3	14	17
	6	6	12	5	11	16
	9	3	12	0	4	4
	6	2	8	0	7	7
	10	2	12	5	5	10
	4	0	4	1	3	4
	3	3	6	3	1	4
	7	3	10	11	4	15
	3	10	13	7	9	16
	6	5	11	6	6	12
	4	7	11	7	5	12
	5	4	9	9	5	14
	5	2	7	3	10	13
	6	1	7	5	10	15
	5	2	7	7	8	15
	4	7	11	10	7	17
	9	3	12	8	8	16
	9	4	13	6	8	14
	6	8	14	14	5	19
8	6	14	7	9	16	
			11	7	18	
$\Sigma$	131	82	213	133	151	284






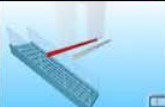


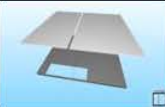



## Příloha 6: Report projektu X – Solibri Model Checker

		<b>Presentation 1</b>	
Model Name	Projekt A SMC Version: 9.6		
Checker	Bc. Daniel Šabart		
Organization	CTU in Prague		
Date	December 10, 2016		
Projekt A IFC ARCH	Date: 2016-11-22 17:27:42 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3		
Projekt A IFC TzB	Date: 2016-11-22 17:27:25 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3		
Projekt A IFC statika	Date: 2016-11-22 17:27:03 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3		


Presentation 1									
Number	Id	Location	Date	Author	Title	Picture	Issue Description	Issue Comment	Status
1	7	(C) Projekt A IFC statika (A) Přízemí, (A) Podlaží 2	10-Dec-2016	DS	Beam, Suspended Ceiling, Light Fixture, Slab too close to Window component		Clearance in Front of Windows	DŠ, 10/12/2016: Okno zasahuje do podlahy 2NP	Open
2	25	(C) Projekt A IFC statika (A) Přízemí, (A) Podlaží 2	10-Dec-2016	DS	Okno vs podhled		Okno zasahuje do podhledu mlstnosti		Open
3	23	(A) Přízemí	10-Dec-2016	DS	Příčka vs okno		Příčka zasahuje do okna		Open
4	9	(A) Přízemí	10-Dec-2016	DS	Stair too close to Window component		Clearance in Front of Windows	DŠ, 10/12/2016: Okno koliduje se schodištěm	Open
5	29	(A) Přízemí, (A) Podlaží 2	10-Dec-2016	DS	Okno podhled		Okno zasahuje do podhledu		Open
6	30	(A) Přízemí, (A) Podlaží 2	10-Dec-2016	DS	podhled vs okno		Podhled zasahuje do okna		Open
7	28	(A) Přízemí, (A) Podlaží 2	10-Dec-2016	DS	Okno vs podlaha		Okno zasahuje do podlahy 2NP		Open
8	11	(A) Přízemí	10-Dec-2016	DS	0610 x 1830mm too close to 0915 x 1500mm 2 component		Clearance in Front of Windows	DŠ, 10/12/2016: Kolize oken	Open
9	12	(A) Přízemí	10-Dec-2016	DS	1500 x 1800mm 2 too close to 1500 x 1800mm 2 component		Clearance in Front of Windows	DŠ, 10/12/2016: Duplikace okna	Open
10	13		10-Dec-2016	DS	(A) Door.0.5 (0762 x 2032mm) and (B) Flow Terminal.0.5 (Undefined) are intersecting		Building Services and Doors and Windows The depth, width, height, and volume of the intersections are: (B) Flow Terminal.0.5, (A) Door.0.5, 38 mm, 25 mm, 380 mm, 0.00 m3	DŠ, 10/12/2016: Kolize vany a dveří	Open
11	14		10-Dec-2016	DS	(A) Wall.0.1 (Interiér, tvárnice zdivo 100) and (B) Flow Terminal.0.5 (Undefined) are intersecting		Building Services and Other Construction Components (Insulations Not Included) The depth, width, height, and volume of the intersections are: (B) Flow Terminal.0.5, (A) Wall.0.1, 60 mm, 41 mm, 380 mm, 0.01 m3	DŠ, 10/12/2016: kolize vany a příčky	Open
12	15		10-Dec-2016	DS	(A) Wall.0.9 (CW 102-50-215p 2 parapet) and (B) Flow Terminal.0.5 (Undefined) are intersecting		Building Services and Other Construction Components (Insulations Not Included) The depth, width, height, and volume of the intersections are: (B) Flow Terminal.0.5, (A) Wall.0.9, 26 mm, 26 mm, 380 mm, 0.01 m3	DŠ, 10/12/2016: kolize vany a stěny	Open
13	16		10-Dec-2016	DS	CW 102-50-215p 2 parapet and Standardní		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 10/12/2016: zařizovací předmět ve stěně	Open


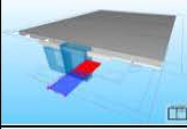
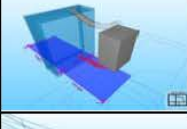
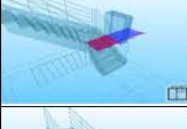
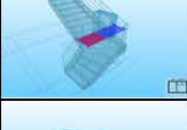

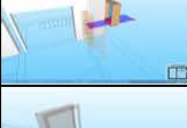
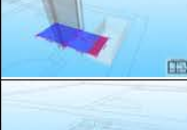




14	17	(A) Přizemí	10-Dec-2016	DŠ	(A) Wall.0.4 (Interiér, tvámicové zdivo 190) and (A) Wall.0.8 (Interiér, tvámicové zdivo 190) are inside head clash		Wall - Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.0.4, (A) Wall.0.8, 2.938 m, 214 mm, 2.600 m, 1.63 m <sup>3</sup>	DŠ, 10/12/2016: Duplikace stěny	Open
15	18	(A) Přizemí	10-Dec-2016	DŠ	(A) Wall.0.1 (Interiér, tvámicové zdivo 100) and (A) Wall.0.7 (CW 102-50-21Sp 2 parapet) are intersecting		Wall - Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.0.7, (A) Wall.0.1, 190 mm, 124 mm, 2.600 m, 0.06 m <sup>3</sup>	DŠ, 10/12/2016: Příčka zasahuje do nosné stěny	Open
16	19	(A) Přizemí	10-Dec-2016	DŠ	(A) Wall.0.2 (Interiér, tvámicové zdivo 100) and (A) Wall.0.4 (Interiér, tvámicové zdivo 190) are intersecting		Wall - Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.0.4, (A) Wall.0.2, 214 mm, 124 mm, 2.600 m, 0.07 m <sup>3</sup>	DŠ, 10/12/2016: Příčka koliduje s jinou příčkou	Open
17	20	(A) Podlaží 2	10-Dec-2016	DŠ	Intersection (A) Roof.2.1 (Zateplená střecha, beton 2) and (A) Wall.1.1 (Interiér, tvámicové zdivo 100) are intersecting		Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.1.1, (A) Roof.2.1, 5.242 m, 124 mm, 273 mm, 0.18 m <sup>3</sup>	DŠ, 10/12/2016: Příčka zasahuje do střechy	Open
18	21	(A) Přizemí	10-Dec-2016	DŠ	(A) Stair.0.1		13.3 Staircase landings	DŠ, 10/12/2016: Málo prostoru nad schody	Open
19	22	(C) Projekt A IFC statika	10-Dec-2016	DŠ	(A) Stair.0.1: 1.100 m		13.4 Head clearance The head clearance height above the stair (A) Stair.0.1 is 1.100 m. The minimum height for the stairs is 2.100 m.	DŠ, 10/12/2016: 3 nosníky a 1 příčka = 4 kolize	Open
20	34	(C) Projekt A IFC statika (A) Přizemí, (A) Podlaží 2	10-Dec-2016	DŠ	SDK vs nosníky		SDK pohledy v kolizi s nosníky	DŠ, 10/12/2016: 7x kolize	Open
21	35	(A) Přizemí	10-Dec-2016	DŠ	SDK pohledy nízké		Nízká světlá výška	DŠ, 10/12/2016: 2x soft kolize	Open
22	36	(A) Podlaží 2	10-Dec-2016	DŠ	SDK světlá výška		Malá světlá výška	DŠ, 10/12/2016: 2x kolize (soft)	Open
23	37	(C) Projekt A IFC statika (A) Přizemí	10-Dec-2016	DŠ	Nosníky v příčkách			DŠ, 10/12/2016: celkem 12 kolizí - neřešen v prostupy	Open


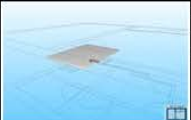





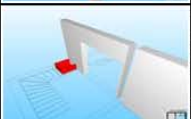








## Příloha 7: Report projektu Y – Solibri Model Checker

		Projekt Y report
Model Name	Projekt B Arch Version: 9.6	
Checker	Bc. Daniel Šabart	
Organization	CTU in Prague	
Date	December 11, 2016	
Projekt B Arch	Date: 2016-12-11 19:31:53 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3	
Projekt B TZB	Date: 2016-12-11 19:29:07 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3	
Projekt B statika	Date: 2016-12-11 19:33:37 Application: Autodesk Revit 2017 (CSY) IFC: IFC2X3	

Presentation 1					
Number	Id	Issue Title	Picture	Issue Description	Issue Comment
1	1	Wrong value of Property - Length: 300 mm		Column Length Column component(s) have wrong value. The actual value of Property: Length is 300 mm. ≥ 500 mm.	
2	2	Systém konstrukčních nosníků: Systém rámové konstrukce, UB305x165x40, Jednoduché, Beton, litý na místě, 225mm, 0915 x 1220mm too close to 0915 x		Clearance in Front of Windows	DŠ, 11/12/2016: 5
3	3	Flow Terminal, Wall too close to Window component		Clearance in Front of Windows	
4	4	Standardní, Výchozí too close to 0915 x 1220mm component		Clearance in Front of Windows	
5	5	190mm max. šířka schodu podstupnice, 250mm too close to 0915 x 1220mm component		Clearance in Front of Windows	
6	6	190mm max. šířka schodu podstupnice, 250mm too close to 0915 x 1220mm component			
7	7	0915 x 1500mm 2 too close to 0915 x 1220mm component		Clearance in Front of Windows	
8	8	Furniture, Wall, Window too close to Window component		Clearance in Front of Windows	
9	9	Furniture too close to Door component		Clearance in Front of Doors	
10	10	(A) Window.0.11 (0915 x 1220mm) and (B) Object.1.55 (Výchozí) are intersecting		Building Services and Doors and Windows The depth, width, height, and volume of the intersections are: (B) Object.1.55, (A)	
11	11	(A) Stair.0.1 (190mm max. šířka schodu podstupnice, 250mm) and (B) Flow Terminal.0.1 (Undefined) are intersecting		Building Services and Other Construction Components (Insulations Not Included) The depth, width, height, and volume of the intersections are:	



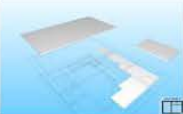
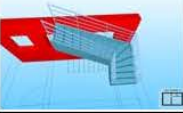

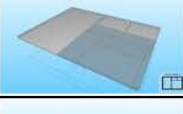

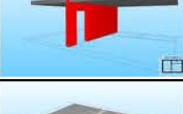

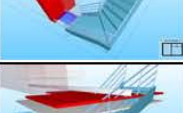
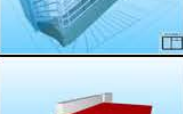





12	12	(A) Suspended Ceiling.0.2 (Jednoduché)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 3
13	13	(A) Suspended Ceiling.0.8 (Jednoduché)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 2
14	14	(A) Wall.0.13 (Interiér, tvárniceové zdivo 100)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 2
15	15	(A) Wall.0.14 (Interiér, tvárniceové zdivo 100)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 2
16	16	Interiér, tvárniceové zdivo 100 and Výchoz		Building Services and Other Construction Components (Insulations Not Included)	
17	17	(A) Wall.0.15 (Interiér, tvárniceové zdivo 100)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 4
18	18	(A) Wall.0.16 (Interiér, tvárniceové zdivo 100)		Building Services and Other Construction Components (Insulations Not Included)	
19	19	(A) Wall.0.17 (Interiér, tvárniceové zdivo 250) and (B) Flow Terminal.0.1 (Undefined) are intersecting		Building Services and Other Construction Components (Insulations Not Included) The depth, width, height, and volume of the intersections are:	
20	20	(A) Wall.0.2 (CW 102-50-215p 2 parapet)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 5
21	21	(A) Wall.0.4 (CW 102-50-215p 2 parapet)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 4
22	22	(A) Wall.0.7 (Interiér, tvárniceové zdivo 100)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 2
23	23	(A) Wall.0.8 (Interiér, tvárniceové zdivo 100)		Building Services and Other Construction Components (Insulations Not Included)	
24	24	(A) Wall.0.9 (Interiér, tvárniceové zdivo 190)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 2
25	25	(A) Wall.1.1 (Interiér, tvárniceové zdivo 190)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 2





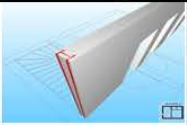
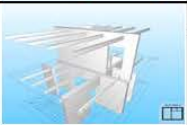
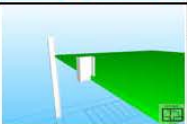
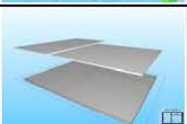
26	26	(A) Wall.1.2 (Interiér, tvárnivé zdivo 100)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 2
27	27	(A) Wall.1.5 (CW 102-50-215p 2 parapet)		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 2
28	28	(A) Wall.0.5 (Interiér, tvárnivé zdivo 100)		Building Services and Other Construction Components (Insulations Not Included)	
29	29	Flow Terminal too close to 0864 x 2032mm component		Distance Between Doors and MEP components	
30	30	Standardní, Výchozí too close to 0864 x 2032mm component		Distance Between Doors and MEP components	
31	31	Intersections of Assembly		Building Services and Other Construction Components (Insulations Not Included)	DŠ, 11/12/2016: 3
32	32	Components too close		Distance Between Columns/Beams and MEP components	DŠ, 11/12/2016: 2
33	33	(C) Podlaží 2		Distance Between Columns/Beams and MEP components	DŠ, 11/12/2016: 2
34	34	(C) Column.0.1, 90 mm		Components Above Columns (C) Column.0.1 doesn't touch components above itself. Distance to the nearest is 90 mm.	
35	35	Column Components don't touch below		Components Below Columns	DŠ, 11/12/2016: 2
36	36	(A) Wall.0.16 (Interiér, tvárnivé zdivo 100) and (A) Wall.0.8 (Interiér, tvárnivé zdivo 100) are duplicates		Wall - Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.0.16, (A) Wall.0.8, 124 mm, 124 mm, 2.800 m,	
37	37	(A) Wall.0.13 (Interiér, tvárnivé zdivo 100) and (A) Wall.0.14 (Interiér, tvárnivé zdivo 100) are duplicates		Wall - Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.0.13, (A) Wall.0.14, 264 mm, 124 mm, 2.800 m,	
38	38	(A) Wall.0.2 (CW 102-50-215p 2 parapet) and (A) Wall.0.6 (Interiér, tvárnivé zdivo 100) are intersecting		Wall - Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.0.2, (A) Wall.0.6, 277 mm, 124 mm, 2.800 m,	
39	39	(A) Window.0.2 (0915 x 1220mm) and (A) Window.0.7 (0915 x 1220mm) are intersecting		Window - Window Intersections The depth, width, height, and volume of the intersections are: (A) Window.0.7, (A)	





40	40	Duplicate Components		Suspended Ceiling - Suspended Ceiling Intersections	DŠ, 11/12/2016: 4
41	41	(A) Stair.0.1 (190mm max. šířka schodu podstupnice, 250mm) and (A) Wall.0.1 (CW 102-50-215p 2 parapet) are intersecting		Stair Intersections The depth, width, height, and volume of the intersections are: (A) Stair.0.1, (A) Wall.0.1, 467 mm, 40 mm, 2.037 m,	
42	42	(A) Stair.0.1 (190mm max. šířka schodu podstupnice, 250mm) and (A) Wall.0.2 (CW 102-50-215p 2 parapet) are intersecting		Stair Intersections The depth, width, height, and volume of the intersections are: (A) Stair.0.1, (A) Wall.0.2, 1.551 m, 50 mm, 1.163 m,	
43	43	Intersections Between Roof and Suspended Ceiling		Suspended Ceiling Intersections kolize pohledů a střechy	DŠ, 11/12/2016: 5
44	44	(A) Roof.2.1 (Zateplená střecha, beton 2) and (A) Wall.1.3 (Interiér, tvárnice zdivo 100) are intersecting		Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.1.3, (A) Roof.2.1, 3.764 m, 124 mm, 273 mm,	
45	45	(A) Roof.2.1 (Zateplená střecha, beton 2) and (A) Wall.1.1 (Interiér, tvárnice zdivo 190) are intersecting		Wall Intersections The depth, width, height, and volume of the intersections are: (A) Wall.1.1, (A) Roof.2.1, 5.242 m, 214 mm, 273 mm,	
46	46	Intersections Between Slab and Wall		Wall Intersections	DŠ, 11/12/2016: 4
47	47	(A) Stair.0.1		13.3 Staircase landings Not enough space at the beginning of (A) Stair.0.1. - Component (A) Wall.0.17 intersects the required free area with dimensions 600	DŠ, 11/12/2016: 2
48	48	(A) Stair.0.1: 512 mm		13.4 Head clearance The head clearance height above the stair (A) Stair.0.1 is 512 mm. The minimum height for the stairs is 2.100 m.	DŠ, 11/12/2016: 4
49	49	(A) Suspended Ceiling.0.1 (jednoduché) and (A) Wall.0.17 (Interiér, tvárnice zdivo 250) are intersecting		Suspended Ceiling Intersections The depth, width, height, and volume of the intersections are: (A) Suspended Ceiling.0.1,	
50	50	Intersections Between Assembly and Suspended Ceiling		Nosníky ve stěnách a podhledech	DŠ, 11/12/2016: 5
51	51	Intersections Between Beam and Suspended Ceiling		Nosníky ve stěnách a podhledech	DŠ, 11/12/2016: 2
52	52	Intersections Between Beam and Wall		Nosníky ve stěnách a podhledech	DŠ, 11/12/2016: 9
53	53	(A) Window.0.1 (0915 x 1500mm 2) and (C) Beam.0.6 (UB305x165x40) are intersecting		Nosníky ve stěnách a podhledech The depth, width, height, and volume of the intersections are: (C) Beam.0.6, (A)	



54	54	(A) Wall.0.9 (Interiér, tvárnice zdivo 190) and (C) Column.0.1 (UC305x305x97) are inside each other		Nosníky ve stěnách a podhledech The depth, width, height, and volume of the intersections are: (C) Column.0.1, (A) Wall.0.9,	
55	55	(A) Wall.0.1 (CW 102-50-215p 2 parapet) and (C) Column.0.2 (UC305x305x97) are intersecting		Nosníky ve stěnách a podhledech The depth, width, height, and volume of the intersections are: (C) Column.0.2, (A) Wall.0.1,	
56	56	(A) Wall.0.9 (Interiér, tvárnice zdivo 190) and (C) Column.0.2 (UC305x305x97) are intersecting		Nosníky ve stěnách a podhledech The depth, width, height, and volume of the intersections are: (C) Column.0.2, (A) Wall.0.9,	
57	57	Intersections Between Assembly and Wall		Nosníky ve stěnách a podhledech	DŠ, 11/12/2016: 11
58	58	Intersections Between Column and Suspended Ceiling		Nosníky ve stěnách a podhledech	DŠ, 11/12/2016: 2
59	59	Too Low Distance		Světelná výška místnosti	DŠ, 11/12/2016: 2