

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ

ÚSTAV ŘÍZENÍ A EKONOMIKY PODNIKU



OPTIMALIZACE USPOŘÁDÁNÍ POLOŽEK VE  
VÝROBNÍM SKLADU

OPTIMIZATION OF ITEMS LAYOUT IN A PRODUCTION WAREHOUSE

DIPLOMOVÁ PRÁCE

AUTOR: Bc. JOSEF POJŽÁREK  
VEDOUCÍ PRÁCE: Ing. PAVEL SCHOLZ  
STUDIJNÍ PROGRAM: ŘÍZENÍ PRŮMYSLOVÝCH SYSTÉMŮ

PRAHA, 2024

## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Pojžárek** Jméno: **Josef** Osobní číslo: **483358**  
Fakulta/ústav: **Fakulta strojní**  
Zadávající katedra/ústav: **Ústav řízení a ekonomiky podniku**  
Studijní program: **Řízení průmyslových systémů**  
Specializace: **Bez specializace**

## II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

**Optimalizace uspořádání položek ve výrobním skladu**

Název diplomové práce anglicky:

**Optimization of Items Layout in a Production Warehouse**

Pokyny pro vypracování:

1. Úvod – zdůvodnění zadání, cíle a úkoly práce
2. Teoretická část – logistika, skladování a druhy skladů, počítačová simulace, její přednosti, omezení a oblasti aplikace, přehled simulačních nástrojů, simulační projekt a jeho kroky
3. Praktická část – charakteristika řešeného problému, tvorba koncepčního a simulačního modelu, simulace a analýza současného stavu, návrh optimalizačního řešení a jeho simulace, vyhodnocení výsledků
4. Závěry a doporučení

Seznam doporučené literatury:

1. JUROVÁ, Marie a kol. Výrobní a logistické procesy v podnikání. Praha: Grada Publishing, 2016. ISBN 978-80-247-5717-9.
2. GROS, Ivan a kol. Velká kniha logistiky. Praha: Vysoká škola chemicko-technologická v Praze, 2016. ISBN 978-80-7080-952-5.
3. MANLIG, František. Využití počítačové simulace výrobních systémů. Liberec: Technická univerzita v Liberci, 2014. ISBN 978-80-7494-162-7.
4. BANKS, Jerry, ed. Handbook of simulation: principles, methodology, advances, applications, and practice. New York: John Wiley & Sons, 1998. ISBN 0-471-13403-1.

Jméno a pracoviště vedoucí(ho) diplomové práce:

**doc. Ing. Jan Horejc, Ph.D. ústav řízení a ekonomiky podniku FS**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

**Ing. Pavel Scholz ústav řízení a ekonomiky podniku FS**

Datum zadání diplomové práce: **03.04.2024** Termín odevzdání diplomové práce: **21.07.2024**

Platnost zadání diplomové práce: **28.02.2025**

\_\_\_\_\_  
doc. Ing. Jan Horejc, Ph.D.  
podpis vedoucí(ho) práce

\_\_\_\_\_  
Ing. Miroslav Žilka, Ph.D.  
podpis vedoucí(ho) ústavu/katedry

\_\_\_\_\_  
doc. Ing. Miroslav Španiel, CSc.  
podpis děkana(ky)

## III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací.  
Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta

## Prohlášení:

Prohlašuji, že jsem svoji diplomovou práci vypracoval samostatně. Dále prohlašuji, že jsem všechny použité zdroje správně a úplně citoval a uvádím je v příloženém seznamu použité literatury.

Nemám závažný důvod proti zpřístupnění této závěrečné práce ve smyslu §60 zákona č. 121/2000Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) v platném znění. Souhlasím také s případnou publikací výsledků diplomové práce nebo její podstatné části, pokud budu uveden jako její spoluautor.

V Praze dne:

Podpis:



## **Poděkování:**

Tímto bych chtěl poděkovat svému vedoucímu diplomové práce Ing. Pavlu Scholzovi za vedení, ochotu a cenné rady a připomínky k mé práci. Dále bych rád poděkoval pracovníkům ze společnosti AIMTEC a.s. a pracovníkům společnosti Safran Cabin CZ s.r.o. za ochotu, trpělivost a hodnotné rady. V neposlední řadě pak své rodině a přátelům za pomoc a morální podporu při psaní této práce a po celou dobu studia.

# Abstrakt

Tato diplomová práce se zabývá simulační studií výrobního skladu v závodě společnosti Safran Cabin. Cílem práce bylo vytvořit simulační studii a navrhnout varianty optimalizace rozmístění položek ve skladu. Simulační model byl vytvořen v softwaru FlexSim 2023. Vyhodnocení výsledků probíhalo na základě porovnání nově navržených variant se současným stavem. Výsledky simulace ukázaly, že navržená opatření povedou ke zkrácení času dokončení setu objednávek o 0,23 h/set, snížení průměrného času vyskladnění dílu z pozice o 0,64 s/pozici, snížení počtu dílů v neergonomických pozicích o 94 ks a zkrácení průměrné vzdálenosti uražené pracovníkem o 416,97 m. Počet objednávek zpracovaných za jeden týden se zvýšil o 18 oproti současnému stavu. V závěru práce jsou uvedeny další doporučení vedoucí ke zlepšení efektivity skladového procesu.

## Klíčová slova

simulace, simulační studie, optimalizace, FlexSim, skladování, logistika

# Abstract

This diploma thesis deals with a simulation study of the production warehouse at the Safran Cabin plant. The goal of the work was to create a simulation study and propose options for optimizing the distribution of items in the warehouse. The simulation model was created in the FlexSim 2023 software. The results were evaluated based on a comparison of the newly designed variants with the current state. The results of the simulation showed that the proposed measures will lead to a reduction in the time to complete the set of orders by 0.23 h/set, a reduction in the average time of removing a part from a position by 0.64 s/position, a reduction in the number of parts in non-ergonomic positions by 94 pcs and a reduction in the average distance offended by the worker by 416.97 m. The number of orders processed in one week increased by 18 compared to the current state. At the end of the thesis, further recommendations are given leading to the improvement of the efficiency of the warehouse process.

## Key words

simulation, simulation study, optimization, FlexSim, storage, logistics

# Obsah

Abstrakt .....	5
Klíčová slova .....	5
Abstract .....	5
Key words .....	6
Úvod .....	13
1    Logistika .....	15
1.1    Logistické systémy a funkce .....	15
1.2    Výrobní logistika .....	17
1.3    Materiálový tok a jeho analýza .....	17
1.3.1    Spaghetti diagram .....	18
1.3.2    Sankeyův diagram .....	19
1.3.3    Postupový diagram .....	20
1.3.4    Value Stream Mapping .....	21
1.4    Optimalizace v logistice .....	22
2    Skladové systémy .....	24
2.1    Funkce skladů .....	24
2.2    Druhy skladů .....	25
2.2.1    Sklady s manuální obsluhou .....	25
2.2.2    Sklady s automatizovanou obsluhou .....	26
2.3    Přínosy a náklady skladů .....	32
2.4    Optimalizace ve skladech .....	33
2.4.1    Ukazatele výkonnosti ve skladech .....	34



2.4.2	Optimalizace technické struktury .....	35
2.4.3	Optimalizace provozní struktury.....	37
2.4.4	Optimalizace řízení skladu .....	39
2.4.5	Využití simulací při optimalizaci.....	39
3	Simulace .....	40
3.1	Historie simulací.....	40
3.2	Dělení simulací .....	41
3.2.1	Diskrétní simulace.....	41
3.2.2	Spojité simulace .....	42
3.2.3	Deterministická simulace.....	42
3.2.4	Stochastická simulace .....	43
3.3	Základní vlastnosti simulačních SW .....	43
3.4	Popis vybraných simulačních SW.....	44
3.4.1	Arena.....	44
3.4.2	Plant Simulation .....	45
3.4.3	FlexSim .....	46
3.4.4	Visual Components .....	47
3.5	Využití simulací .....	47
3.6	Přednosti a omezení simulací .....	49
3.6.1	Přednosti.....	49
3.6.2	Omezení .....	50
4	Simulační projekt .....	51





4.1	Vymezení projektu .....	52
4.2	Tvorba a testování modelu .....	52
4.2.1	Definice struktury .....	52
4.2.2	Sběr a analýza dat .....	53
4.2.3	Tvorba modelu .....	53
4.2.4	Verifikace .....	54
4.2.5	Validace .....	54
4.3	Experimentování a optimalizace .....	55
4.4	Vyhodnocení projektu .....	56
5	Úvod do praktické části .....	57
5.1	Představení společnosti .....	57
6	Popis použitého simulačního nástroje .....	59
6.1	Uživatelské rozhraní .....	59
6.2	Terminologie .....	60
6.2.1	Objects .....	60
6.2.2	Process Flow Objects .....	61
6.2.3	Labels .....	62
6.2.4	Lists .....	62
6.2.5	Tables .....	62
6.3	Základní objekty 3D modelu .....	62
6.3.1	Source .....	62
6.3.2	Queue .....	63



6.3.3	Task Executer .....	63
6.3.4	Operator.....	64
6.3.5	Rack.....	64
6.4	Základní objekty Process flow.....	65
6.4.1	Source .....	65
6.4.2	Assign Labels .....	65
6.4.3	Create Tokens .....	66
6.4.4	Custom Code.....	66
6.4.5	Decide .....	66
6.4.6	Zone .....	67
6.4.7	Resource .....	67
6.4.8	Objects .....	68
6.4.9	Delay .....	68
6.4.10	Task Sequences.....	68
6.4.11	Warehousing.....	69
6.4.12	Sub – flow .....	69
7	Vymezení projektu.....	70
8	Koncepční model a sběr dat .....	71
8.1	Rozvržení skladu .....	71
8.2	Definice procesu .....	73
8.3	Sběr dat.....	74
9	Tvorba simulačního modelu .....	76



9.1	Model č.1 – 3D model a Process flow.....	76
9.2	Model č.2 – 3D model.....	78
9.2.1	Sklad.....	78
9.2.2	Pracovníci.....	79
9.2.3	Žebříky.....	80
9.2.4	A* Navigator .....	81
9.3	Model č.2 – model Process Flow .....	82
9.3.1	Orders creation .....	84
9.3.2	Orders prepair and picking .....	86
9.3.3	Order Sub Flow .....	88
9.3.4	End of order picking process .....	96
9.3.5	Verifikace .....	97
9.3.6	Validace.....	98
10	Experimentální část .....	99
10.1	Vstupní data.....	99
10.2	Současný stav.....	100
10.1	Varianta 1.....	101
10.2	Varianta 2.....	102
11	Vyhodnocení výsledků .....	103
11.1	Porovnání současného stavu a varianty 1 .....	103
11.2	Porovnání současného stavu a varianty 2 .....	104
11.3	Porovnání varianty 1 a varianty 2 .....	104



---

11.4	Závěrečné vyhodnocení .....	105
	Závěr .....	106
	Seznam obrázků .....	108
	Seznam tabulek .....	113
	Seznam použit literatury .....	114

## Úvod

S problémem rostoucí konkurence a s tím spojenými rostoucími nároky na udržení se na trhu, se ve svém odvětví dnes potýká ne jeden podnik. V podnicích jsou tak kladeny stále větší a větší požadavky, jak na výrobu, tak logistiku, vedoucí k zajištění jejich efektivity a udržitelnosti. S tímto problémem jim v moderní době pomáhají různé počítačové softwary. Pomocí nich lze přesněji plánovat a řídit procesy v podniku. Jedním z nejmodernějších nástrojů jsou simulační softwary, pomocí kterých se vytvářejí simulační modely. Jelikož dostupnost těchto softwarů roste, tak se stále častěji používají k optimalizaci procesů, čímž dochází k poklesu nákladů na optimalizaci. Tato opatření mohou firmám poskytnout konkurenční výhodu v důsledku obrovského mezinárodního konkurenčního prostředí v oblasti výroby a logistiky.

Tato diplomová práce pojednává o použití simulačního softwaru FlexSim pro potřebu optimalizace výrobního skladu v jednom ze závodů společnosti Safran Cabin. Tato společnost se zabývá výrobou vybavení interiérů (kávovarů, úložných skříněk, toalet aj.) zejména pro letadla značky Airbus, či luxusní soukromá letadla. Cílem této práce je na základě poskytnutých dat vytvořit simulační model skladu a navrhnout vhodné optimalizační řešení, které by zkrátilo dobu přípravy a výdeje dílů. Výsledný simulační model pak bude sloužit jako nástroj pro další optimalizaci daného skladu.

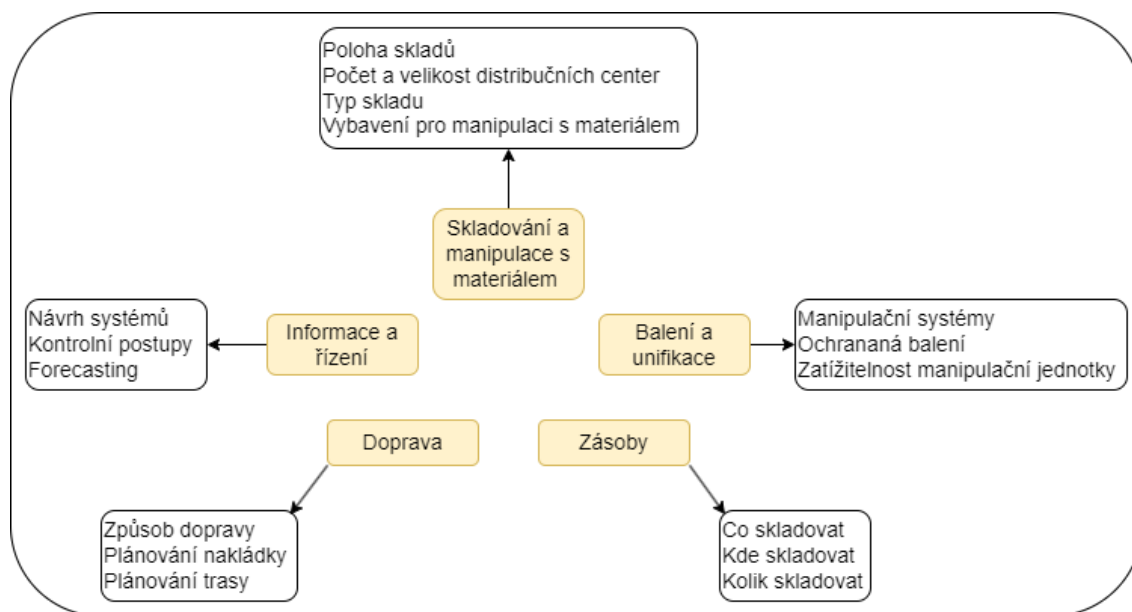
V teoretické části se práce zabývá logistikou, skladovými systémy a simulacemi. V rámci logistiky jsou popsány základní logistické systémy a funkce, dále výrobní logistika, analýza materiálového toku, a nakonec možnosti její optimalizace. Další kapitola je věnována funkcím a druhům skladových systémů včetně jejich přínosů a nákladů pro podnik. Jsou zde popsány také možnosti jejich optimalizace, a to jak z hlediska technické a provozní struktury, tak i jejich řízení. Následná kapitola pojednává o historii simulací, jejich rozdělení, využití, přednostech a omezeních. Dále jsou zde popsány základní vlastnosti simulačních softwarů a také vybrané simulační softwary. Teoretická část je zakončena kapitolou o simulačním projektu, kde je krok po kroku popsána jeho tvorba.



Praktická část práce se zabývá vlastní simulační studií. Nejprve je krátce představena společnost Safran Cabin a dále jsou zde vysvětleny základní pojmy potřebné pro práci se zvoleným simulačním softwarem FlexSim. Samotná simulační studie je dále zpracována po jednotlivých krocích popsaných v teoretické části. Na jejím počátku jsou rozebrány cíle řešení a dále je popsán koncepční model a způsob sběru dat. Na koncepční model navazuje tvorba simulačního modelu. Ten je zde dopodrobna vysvětlen, včetně jeho omezení a odchylek od reálného systému. Praktickou část uzavírají kapitoly věnující se experimentům prováděným na vytvořeném modelu a vyhodnocení jejich výsledků.

# 1 Logistika

Logistika byla jako samostatná a významná oblast uznána teprve na konci minulého století, ale její složky: zásoby, doprava a skladování vytvářejí základy průmyslu a hospodářství už mnohem déle. Jedná se o oblast, která se skládá z mnoha dílčích funkcí a podsystémů, ale zároveň je každý z nich brán jako samostatně řízená operace. I to je jedním z hlavních důvodů proč její uznání jako samostatného celku trvalo dlouhou dobu. K tématu logistiky se v dnešní době přistupuje již více vědecky, k čemuž pomohlo také to, že si lidé více uvědomují rozsah a význam logistiky i dodavatelských řetězců. Klíčové komponenty logistiky jsou pro každý podnik trochu rozdílné. Mezi obecně nejpoužívanější se řadí: informace a řízení, doprava, skladování a manipulace s materiálem, balení a unifikace a zásoby. Na Obr. 1 jsou zobrazeny detailnější aspekty jednotlivých komponent. [1]

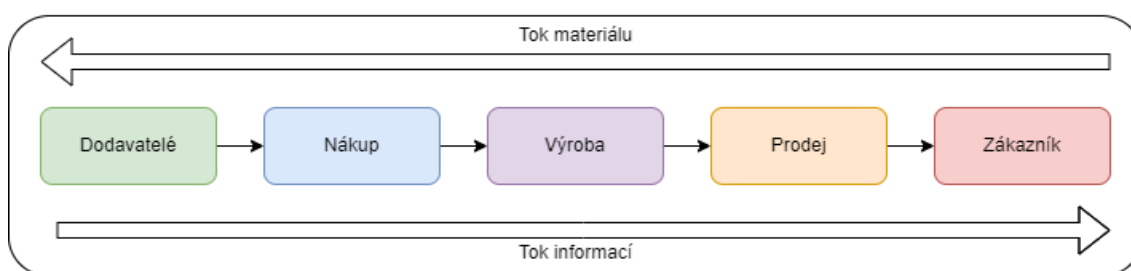


Obr. 1 - Klíčové komponenty distribuce a logistiky

## 1.1 Logistické systémy a funkce

Jak již bylo uvedeno v úvodu této kapitoly, logistika se skládá z několika systémů, které tvoří synergický celek. Různé podniky mohou používat rozdílné systémy např.

dopravní systém, výrobní systém, skladovací systém, aj., podle toho, jaké je jejich zaměření. Sled jednotlivých systémů se nazývá logistický řetězec (viz. Obr. 2). Každý systém plní svojí roli pomocí jeho funkcí a činností (např. plánování poptávky, řízení stavu zásob, balení, zákaznický servis aj.). Funkce jednotlivých systémů reprezentují procesy, které se v daném systému provádějí, např. nakládka, přeprava, kompletace, kontrola a další. Funkce jednoho systému podporují funkce druhého, čímž je možné docílit lepších výsledků než při plnění těchto funkcí pro každý systém zvlášť. Např. budeme mít systém dopravy a systém skladování. Pokud by každý z nich řešil své aktivity zvlášť, izolovaně, pak nebude zajištěn včasný dovoz surovin do skladu.



Obr. 2 - Logistický řetězec

Logistické funkce je možné rozdělit do 4 úrovní:

- Strategické rozhodování – dlouhodobé plánování zdrojů, pravidel a pracovních postupů.
- Dispoziční rozhodování – způsob jakým se uspokojí potřeby zákazníků a podniku.
- Administrativní – informační procesy v logistickém řetězci.
- Operativní – realizace hmotné stránky logistických řetězců.

Konkrétní logistické funkce odvozené z těchto úrovní jsou poté následující:

- Dispoziční – řeší vydávání pokynů a plánů práce v jednotlivých systémech.
- Dopravní – určuje způsob zajištění dopravy uvnitř a vně podniku.
- Překládková – určuje způsob jakým budou výrobky překládány uvnitř a vně podniku.
- Skladová – řeší činnosti související se skladováním.





- Sběrná a distribuční – řeší rozdělování a soustřeďování surovin, polotovarů a hotových výrobků.
- Balící – řeší způsob balení výrobků.
- Manipulační – určuje způsob manipulace se surovinami, polotovary a výrobky.
- Informační – určuje jaké informační zdroje budou použity a způsob získávání, zpracování a využívání informací.
- Řídicí a kontrolní – určuje způsob řízení a kontroly v jednotlivých systémech.
- Racionalizační a inovační – řeší způsoby zlepšování pracovních postupů, skladové úspory a inovace.

A další zvláštní funkce jako např. pojištění, služby pro zákazníky nebo péče o životní prostředí. [2][3]

## 1.2 Výrobní logistika

Výrobní logistika má za úkol zajistit pravidelný přísun materiálu v dostatečném množství k zabezpečení výrobního procesu. Zároveň musí zajistit plynulý odchod již hotových výrobků z výrobního úseku do skladu a následně jejich expedici. Je tedy úzce propojena se zásobovací a odbytovou logistikou.

Mezi hlavní činnosti výrobní logistiky řadíme:

- Operační a mezioperační dopravu
- Mezioperační skladování polotovarů
- Manipulaci s hotovými výrobky
- Zajištění příznivých podmínek pro zaměstnance[2]

## 1.3 Materiálový tok a jeho analýza

Nedílnou součástí podnikové logistiky je také řešení toku materiálu. Jedná se o řízený pohyb materiálu, surovin a polotovarů, který je základem logistických procesů v podniku. S jeho pomocí je možné popsat dynamiku výroby v prostoru a čase. Při



analýze materiálového toku se zkoumá zejména efektivnost pohybu materiálu v rámci výrobního procesu, čímž dochází ke znázornění požadavků výrobních, dopravních, manipulačních a skladovacích procesů a jejich vzájemné vazby.

Tok materiálu je ovlivněn uspořádáním výrobních zařízení, budov a skladů, jejichž vhodným rozvržením lze dosáhnout nezanedbatelných úspor materiálu, času ale i financí.

Průběh a realizace materiálového toku jsou ovlivněny:

- Druhem, typem, objemem a sortimentem výrobního procesu
- Úrovní technologické složitosti výrobního procesu
- Počtem operací v jednotlivých fázích výrobního procesu
- Způsobem řešení dopravy
- Umístěním pomocných provozů a služeb – např. výdejna náradí, středisko údržby apod.

Pro účely analýzy a následného projektování manipulace s materiálem je potřebné rozložení procesu manipulace na menší části: proces, dílčí proces, operace, úkon a pohyb.

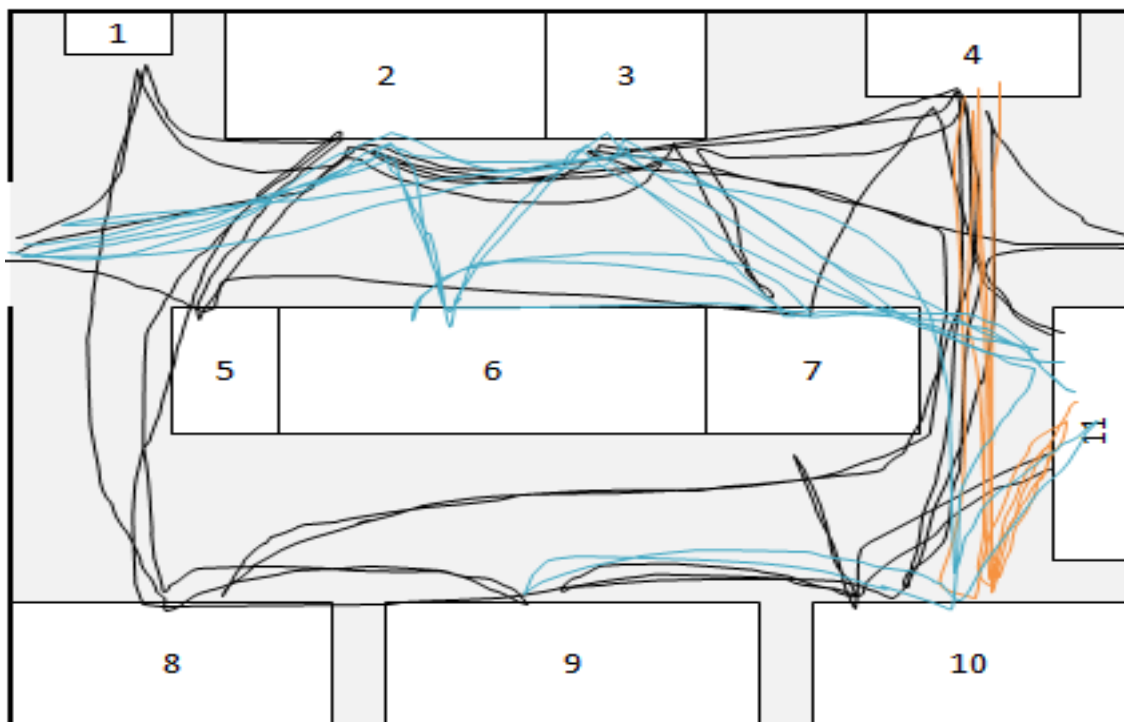
Operace je základní jednotkou manipulace a je možné ji definovat jako záměrnou nepřetržitou změnu polohy předmětu, která je prováděna pracovníkem, manipulačním prostředkem, případně jejich kombinací.

Cílem analýzy materiálového toku je odhalit úzká místa a určit rámeček jejich optimalizace. K analýze se využívají různé druhy diagramů, např. Spaghetti diagram, Sankeyův diagram, Value stream mapping, nebo Postupový diagram. Uvedené diagramy jsou blíže popsány dále v této kapitole. [4]

### 1.3.1 Spaghetti diagram

Tento diagram řadíme mezi jedny z nejjednodušších metod analyzování materiálového toku. Používá se především pro mapování materiálového toku uvnitř podniku a hledání nejhodnější logistické cesty nebo při návrhu rozložení pracoviště.

Každý pohyb pracovníka na určitém pracovišti a v určitém čase je přesně zakreslen do diagramu. Každý přesun a pohyb je odlišen použitím různých barev (např. zbytečná cesta – černě, cesta s materiálem – žlutě). Cesty je možné zaznamenávat do systému pomocí různých elektronických zařízení a softwaru. Příklad diagramu je znázorněn na Obr. 3.[4]



Obr. 3 - Spaghetti diagram, BreezeTree[26]

### 1.3.2 Sankeyův diagram

Jedná se o jeden z nejnámějších a nejméně používaných způsobů, jak vizualizovat materiálový tok napříč podnikem. Sankeyův diagram je založen na principu grafického znázornění struktury a časového průběhu veličiny v daném systému (viz.Obr. 4). Velikost průtoku se přímo úměrně mění s šířkou pásu (čáry). Hlavní oblastí použití jsou

energetické a technologické procesy, v dnešní době je však velmi využíván i v logistice.

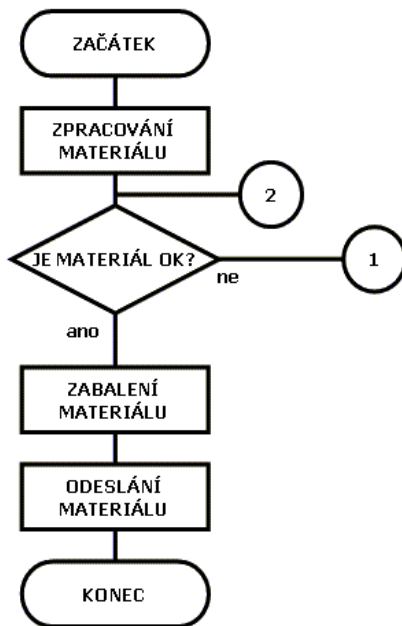
Práce s diagramem je usnadněna použitím různých softwarů.[4]



Obr. 4 - - Sankeyův diagram, The Data School[27]

### 1.3.3 Postupový diagram

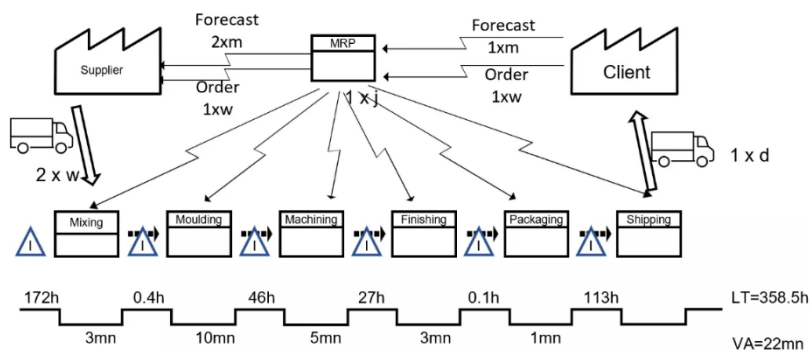
Tento diagram je používán pro znázornění posloupnosti všech manipulačních, technologických a kontrolních operací prováděných na daném výrobku (jednoho, či více) nebo dávce určitého procesu (výrobního i logistického). Procesy jsou popisovány a analyzovány z jejich věcné, časové a nákladové stránky. Postupový diagram lze použít i na nevýrobní procesy a služby. Výstupem diagramu je určení počtu, délky a proporcionality každé operace, což slouží k mapování a přizpůsobení procesů, zejména pak vhodnosti spolupráce mezi pracovišti, případně identifikaci činností zvyšujících hodnotu a zvyšování jejich podílu. Příklad postupového diagramu je na Obr. 5. [4]



Obr. 5 - Postupový (vývojový) diagram, ikvalita

### 1.3.4 Value Stream Mapping

Tato metoda mapování toku materiálu byla vyvinuta společností Toyota v rámci zkoumání štíhlého řízení výroby. Využívá speciální symboly pro popis materiálového i informačního toku. Dále popisuje procesy, které přidávají a nepřidávají hodnotu, z nichž poté vycházejí návrhy na odstranění plýtvání (např. změny ve výrobním procesu, návrh nového výrobního procesu atd.). Výsledkem mapování je identifikace rezerv a plýtvání v procesu. VSM mapa je zobrazena na Obr. 6.[4]



Obr. 6 - VSM mapa, Wevalgo

## 1.4 Optimalizace v logistice

Efektivní řízení logistických procesů je zásadním faktorem pro úspěch a konkurenceschopnost moderních podniků. Optimalizace logistiky zahrnuje komplexní přístup k řízení toku materiálu, informací a financí v rámci celého dodavatelského řetězce.

Jednou z klíčových oblastí optimalizace je efektivní plánování a řízení dopravy. Využitím moderních technologií, jako je např. GPS, umožňuje podniku lépe monitorovat a řídit svůj vozový park. Tím se snižují ujeté kilometry, spotřeba paliva a celkové náklady na dopravu. Dalším důležitým prvkem je sjednocení zásilek a sdílení přepravních kapacit s jinými společnostmi, což vede k vyššímu vytížení vozidel a efektivnějšímu využití zdrojů.

Efektivní řízení skladových procesů a zásob je další klíčovou oblastí optimalizace logistiky. Efektivní rozmístění a uspořádání skladových prostor vede k minimalizaci manipulace se zbožím a snížení přepravních a provozních nákladů. Využití moderních skladových technologií, jako jsou automatizované systémy, dále zvyšuje produktivitu a efektivitu skladových operací. Optimalizace zásob a řízení poptávky pak umožňuje snížit náklady na skladování a vázaný kapitál.

V rámci zvýšení konkurenceschopnosti podniku je důležité se zaměřit především na efektivitu ve výrobních, ale i v logistických procesech. V rámci optimalizace se přistupuje na zavádění štíhlé logistiky. Jejím hlavním cílem je návrh nových a změna stávajících procesů tak, aby došlo k odstranění všech zdrojů plýtvání z procesu a zkrátila se jeho doba. Při zavedení štíhlé logistiky dochází ke zvýšení produktivity a snížení nákladů na logistické procesy. Tato problematika je detailněji rozebrána v Kap.2

Moderní logistika se také neobejde bez efektivního toku informací a využití digitálních technologií. Integrace informačních systémů v rámci dodavatelského řetězce zvyšuje přehlednost a koordinaci procesů. Využití analytických nástrojů pro prediktivní plánování a rozhodování pak umožňuje pružněji reagovat na změny poptávky a optimalizovat logistické procesy. Automatizace administrativních procesů navíc snižuje chybovost a zvyšuje efektivitu celého systému.



Celkově lze říci, že komplexní optimalizace logistických procesů, zahrnující technologické, procesní a organizační změny, je klíčem k dosažení vyšší flexibility, rychlosti a nákladové efektivity v rámci celého dodavatelského řetězce. Podniky, které dokáží efektivně optimalizovat své logistické procesy, získávají významnou konkurenční výhodu na trhu. [1]

## 2 Skladové systémy

Klíčovou součástí při řešení optimalizace logistiky je zefektivnění skladových systémů. Sklady jsou základními objekty pro ukládání výrobků, materiálu, polotovarů apod. Za sklad se považuje jakýkoliv objekt, budova, či prostor, který slouží danému účelu. V této kapitole jsou rozebrány funkce skladů, jejich druhy z hlediska obsluhy a dále také možnosti jejich optimalizace.

### 2.1 Funkce skladů

V historii zastávaly sklady funkci zásobníků, do kterých se ukládaly podle plánu vyrobené výrobky, díly, polotovary, suroviny apod. Z hlediska řízení materiálového toku se jednalo o princip tlaku. Při použití principu tlaku je sklad konečným místem, kam jsou tlačným způsobem umístěny výrobky, vytvořené v předešlých prvcích dodavatelského řetězce, formou zásob. Slouží tedy k uskladnění nadměrné produkce a jeho role je spíše pasivní.

V současné době je funkce skladu mnohem komplexnější. Nově jsou sklady vymezeny jako poskytovatelé vyšších úrovní služeb zákazníkům a činnosti, které se v nich realizují, zvyšují hodnotu pro následujícího partnera v dodavatelském řetězci. Při realizaci dodávek se vychází z požadavků zákazníka a stále více se uplatňuje princip tahu. Většina skladů však pracuje na základě obou principů. Na principu tlaku se podle odhadu budoucí poptávky vytváří pojistná zásoba a na principu tahu se podle objednávek od zákazníků vystavují, kompletují a expedují objednávky dodavatelům.

V rámci řízení materiálového toku vykonávají sklady tyto vyrovnávací funkce:

- Geografická funkce – řeší nerovnováhu v místním rozložení skladů. Využitím vhodně rozmístěných skladů lze např. zkrátit dodací dobu zboží, poskytnout prodej výrobků v blízkosti svých provozů apod., a tím zvýšit konkurenceschopnost výrobce v jednotlivých regionech.



- Sezónní funkce – řeší překlenutí časového rozdílu mezi výrobou a spotřebou sezónních výrobků např. paliv či zemědělských produktů.
- Kapacitní funkce – řeší kapacitní rozdíly mezi jednotlivými prvky dodavatelského systému, např. při nadvýrobě se přebytek uskladní.
- Kompletační (rozdělovací) funkce – řeší skládání (sdružování) zásilek většinou od různých výrobců do ucelených dodávek. Rozdělovací funkce má pak funkci opačnou.
- Pojistná funkce – řeší pojistnou zásobu ve skladech, která umožní dodavateli reagovat na různé výkyvy poptávky. [5]

## 2.2 Druhy skladů

Aby mohli sklady vykonávat své funkce co nejefektivněji, je důležité, aby byly řádně obsluhovány. Sklady lze rozdělit podle stupně automatizace obsluhy na 2 druhy: sklady s manuální obsluhou a dnes, díky trendu automatizace, více používané sklady s automatizovanou obsluhou.

V této podkapitole jsou představeny v praxi nejčastěji používané druhy skladů.

### 2.2.1 Sklady s manuální obsluhou

Ve skladech s manuální obsluhou provádí zaskladňování i vyskladňování výrobků pracovník. Obsluha může být různě mechanizována např. použitím paletových nebo vysokozdvížných vozíků. V rámci skladů s manuální obsluhou rozlišujeme tyto typy skladů: pevné regálové sklady, pojízdné regálové sklady, paletové sklady a víceúrovňové regálové sklady. Jednotlivé typy jsou popsány v následujících odstavcích.

**Pevné regálové sklady** – jsou vybaveny jednoduchými, pevně uchycenými regály bez možnosti pohybu. Poskytují vysokou zatížitelnost a lze je snadno přizpůsobit prakticky jakémukoliv rozměru výrobku.

**Pojízdné regálové sklady** – mají podobnou konstrukci jako pevné sklady, nabízí mnoho možností jejich přestavění a jsou určeny především pro menší sklady. Jejich

základem je kolejnicový systém namontovaný v podlaze a na něm jsou umístěny regály. Výhodou oproti pevnému regálovému systému je značná úspora skladovací plochy. Nevýhodou může být vyšší náročnost manipulace s regály, kvůli jejich váze. Jejich využití se najde převážně v menších provozech s vysokými nároky na úsporu plochy, ale své uplatnění nacházejí i ve větších podnicích.

**Paletové sklady** – využívají, jako hlavní úložnou jednotku pro jednotlivé produkty, palety. Produkty jsou ukládány na paletu ze dřeva (případně plastu, kovu, nebo překližky), která je následně zaskladněna do pevného nebo pojízdného regálového skladu. Manipulace s paletami může být částečně (paletové vozíky) nebo i úplně (vysokozdvížné vozíky) mechanizována, což přispívá k urychlení celého procesu.

**Víceúrovňové regálové sklady** – jsou dvou a více patrové sklady, které mají využití především pro podniky, kde velkou část zásob tvoří produkty o malých rozměrech. Pohyb mezi jednotlivými patry zajišťují schodiště a plošiny pro vysokozdvížné vozíky. [6][7]

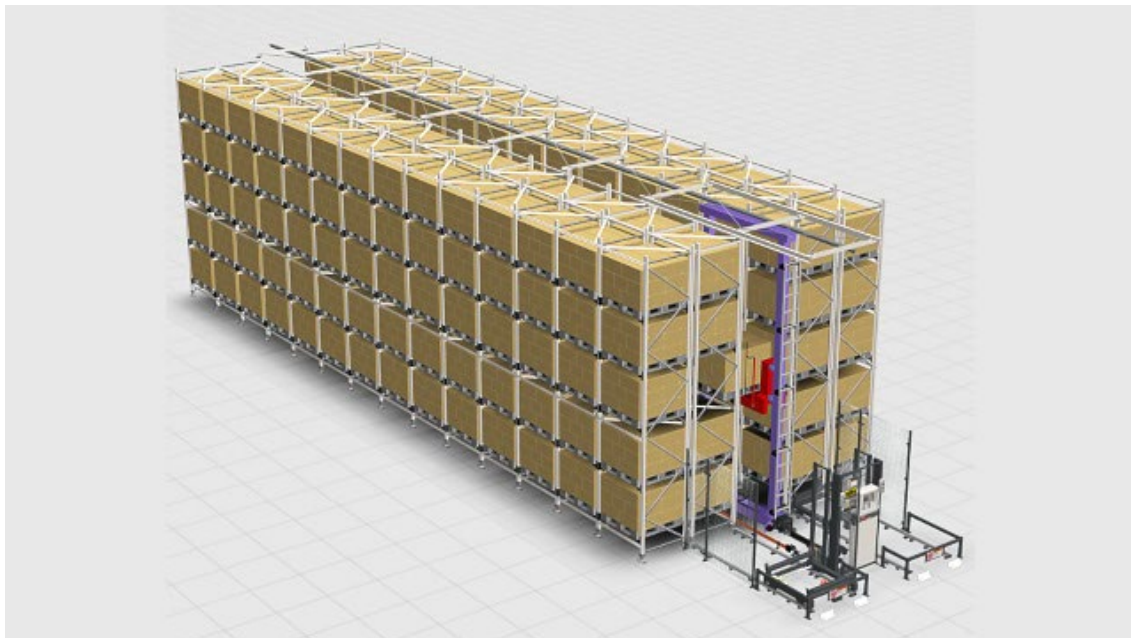
## 2.2.2 Sklady s automatizovanou obsluhou

Se stále se rozvíjícím trendem Průmyslu 4.0 a zvyšující se rychlostí a složitostí všech procesů, přibývá také stále více skladovacích řešení, která jsou schopna na tuto skutečnost reagovat. Jedná se hlavně o sklady s automatizovanou obsluhou, bez přítomnosti člověka. Jejich výhody jsou především ve vyšší rychlosti naskladnění a vyskladnění, úspoře nákladů (zejména personálních) a úspoře prostoru potřebného pro skladování. V rámci automatizovaných skladů rozlišujeme 3 hlavní typy: sklady s automatizovanými zakladači, sklady s autonomními zakladači a robotizované sklady. Jednotlivé typy jsou popsány v následujících odstavcích.

**Sklady s automatizovanými zakladači**, angl. Automated Storage and Retrieval System (zkráceně AS/RS), používají k zaskladnění a vyskladnění produktů automatické manipulátory, jeřáby nebo vysokozdvížné vozíky. Tyto sklady lze dále dělit na:

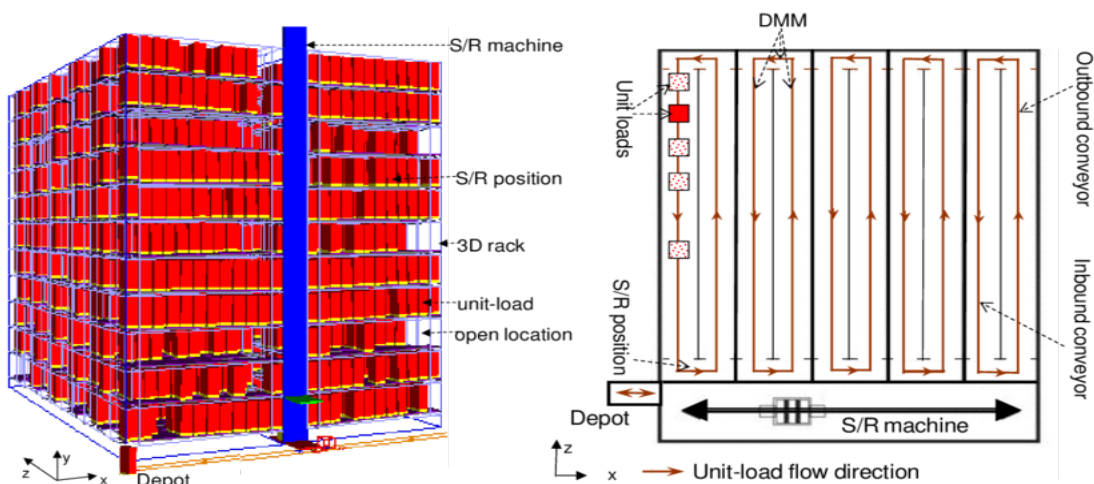
- Systémy hlubinného zakládání – jsou složené ze 2 základních prvků: dopravníků a zakladačů. Dopravníky přepravují produkty od přejímky

k zakladačům. Zakladače obsluhující příslušnou dvojici regálů poté produkty zaskladní. Vyskladňování zboží funguje na obráceném principu. Schéma tohoto typu skladu je zobrazeno na Obr. 7.



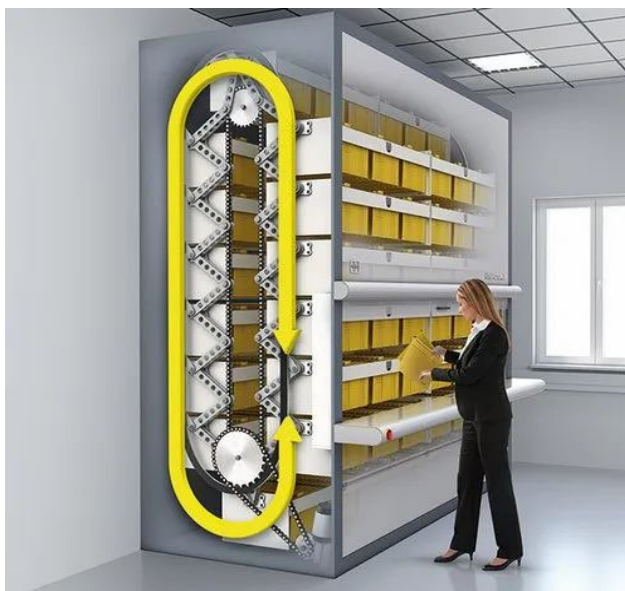
Obr. 7 - Schéma systému hlubinného zakládání, Daifuku

- Kompaktní úložné systémy – jsou velice podobné systémům hlubinného zakládání. Rozdíl je především v rozložení skladu, kdy je každé patro složeno z dvojice dopravníků a zakladač se pohybuje přes tato patra. Schéma tohoto typu skladu je zobrazeno na Obr.8.



Obr. 8 - Schéma kompaktního úložného systému[31]

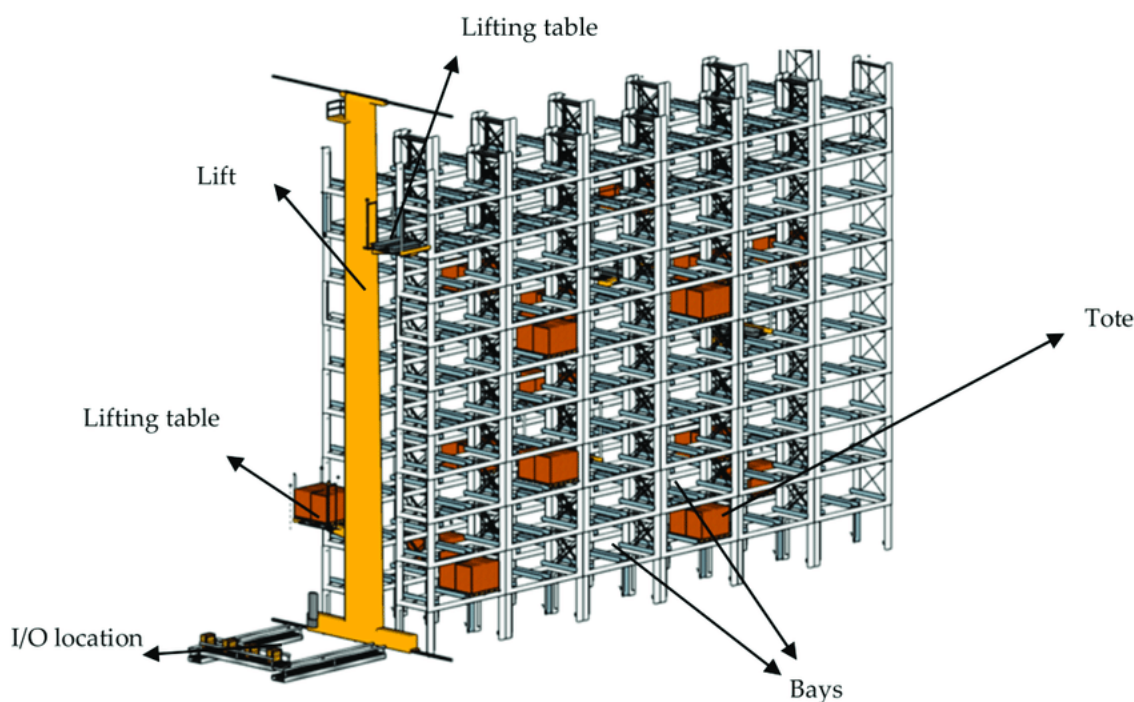
- Karuselové AS/RS systémy – mají všechny regály spojeny do uzavřené smyčky, která vertikálně nebo horizontálně rotuje. Tím je zajištěn přístup zakladače k jednotlivým regálům. Schéma tohoto typu skladu je zobrazeno na Obr. 9.



*Obr. 9 - Schéma karuselového systému, NSR Libya*

**Sklady s autonomními zakladači**, angl Autonomous Vehicle-based Storage and Retrieval System (zkráceně AVS/RS), používají k zaskladnění a vyskladnění produktů autonomní vozíky. Tyto vozíky se pohybují po kolejnicích umístěných v uličkách mezi jednotlivými regály. K nakládání a vykládání produktů slouží výsuvné rameno. Jelikož se vozíky mohou pohybovat pouze po dané úrovni skladu a sami nejsou schopné svislého pohybu, jsou po stranách skladu umístěny výtahy. Výtahy fungují podobně jako zakladače u AS/RS skladu a slouží k přepravě autonomních vozíků do jednotlivých pater skladu. Oproti AS/RS skladu je zde díky pohybu vozíků po celé ploše dané úrovně také výhoda rychlejšího procesu zaskladnění a vyskladnění produktů. V dnešní době existují už AVS/RS sklady které využívají vozíky, které jsou schopné i vertikálního pohybu. Tím

dochází k dalšímu urychlení procesu skladování a zmenšení prostoru potřebného pro sklad. Schéma tohoto skladu je zobrazeno na Obr. 10.



Obr. 10 - Schéma skladu s autonomními zakladači[33]

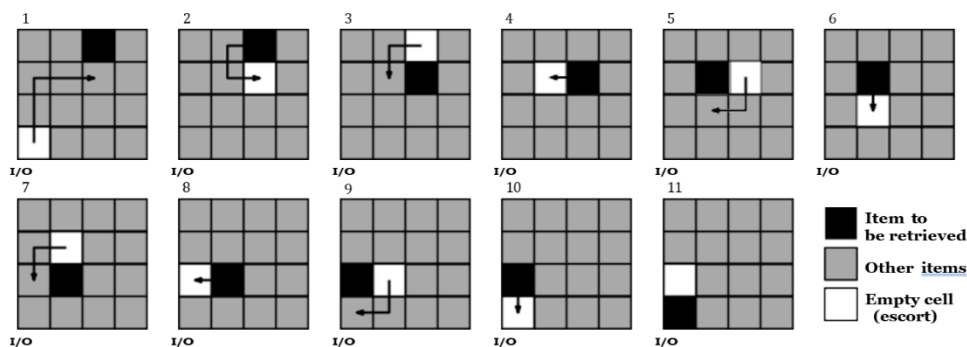
**Robotizované sklady** využívají k přepravě produktů robotické systémy a AGV (Automated Guided Vehicle – automaticky naváděné vozíky). Jejich dalším charakteristickým rysem je snaha o co největší skladovací poměr – skladování co největšího množství výrobků, na co nejmenší ploše. Robotizované sklady rozdělujeme podle způsobu skladování na:

- Puzzlové skladovací systémy – fungují na základě hry amerického šachisty a tvůrce hlavolamů Sama Loyda. Cílem této hry je složit do přesného pořadí 15 náhodně rozmístěných čtvercových bloků v poli o velikosti 4x4 bloky. Schéma hry je zobrazeno na Obr.11.

5	2		13
15	6	9	4
7	3	8	12
11	1	10	14

Obr. 11 – Sam-Loydovo puzzle,  
Murderousmaths

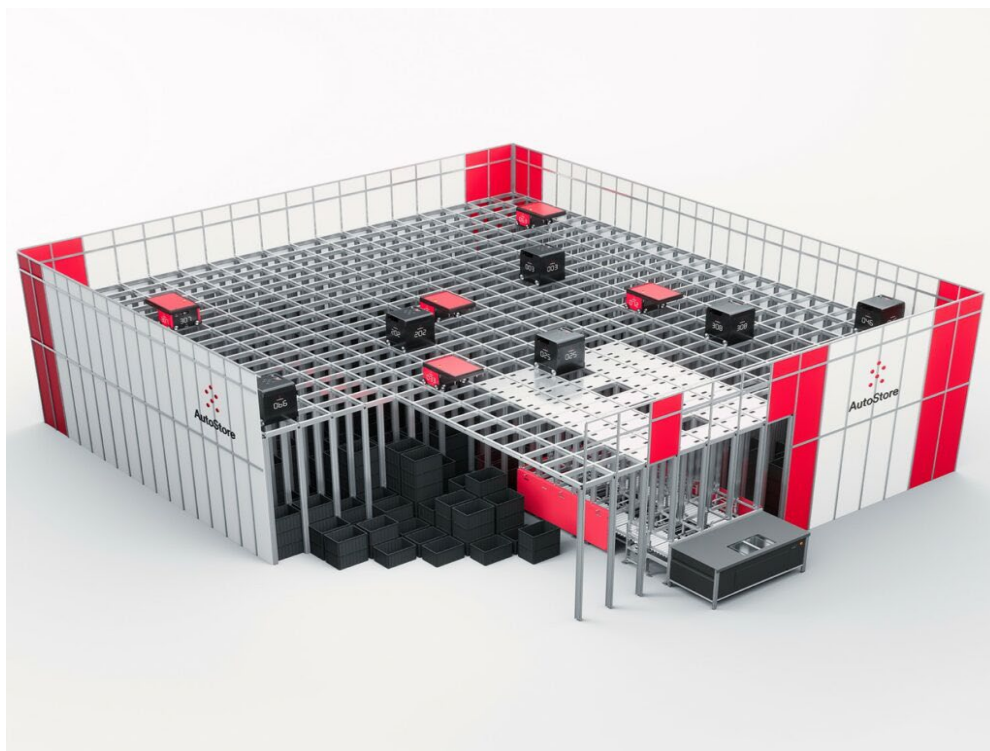
Sam Loydova úloha aplikovaná na skladovací systém může být znázorněna podle Obr. 12. Skladovací systém má rozměry 4x4 s jedním I/O (vstupním/výstupním) portem, kde skladovací hustota je  $(n - 1)/n$ , přičemž  $n$  představuje celkový počet skladovacích buněk. Úkolem je vyskladnit obsah náhodně zvolené buňky (černé čtvercové pole na obrázku).



Obr. 12 - Schéma skladového systému na základě Sam Loydova puzzle[8]

- Robotické skladovací systémy – jsou tvořeny krychlovou sítí skladovacích pozic. Na vrchu této krychle je umístěna rámová konstrukce sloužící pro pohyb robotů. Roboti zaskladňují a vyskladňují zboží tak, že vždy vezmou příslušný vrchní box. Pokud se v něm nalézá požadovaný produkt, pak je box

přemístěn na výdejní pozici. Pokud v něm požadovaný produkt není, pak robot přeskládá boxy na volnou vedlejší pozici, dokud nenajde požadovaný box. Po jeho odvezení obvykle volný robot zaskladní přeskládané boxy zpět na původní pozici. Schéma tohoto typu skladu je zobrazeno na Obr. 13.



*Obr. 13 - Schéma robotického skladovacího systému, Strongpoint*

- Mobilní robotický skladovací systém – se používá zejména při potřebě skladovat mnoho typů drobných výrobků. Tento systém je tvořen odbavovacími stanicemi, robotickými vozítky a regály. Vozítky nepřevážejí samotné produkty, ale celé regály. Výhodou tohoto systému je jeho

jednoduchost, flexibilita a rozšíření. Ukázka tohoto systému je zobrazena na Obr. 14. [8]



Obr. 14 - Ukázka mobilního robotického skladového systému, Warehousenews

## 2.3 Přínosy a náklady skladů

Sklady mají zpravidla dva druhy výhod: úsporu nákladů a zlepšení úrovně služeb zákazníkům.

K úspoře nákladů dochází několika způsoby. Prvním z nich je vytvoření pravidel pro vytváření hromadných objednávek. Díky tomu dochází k plnému vytížení dopravních prostředků a tím ke snížení přepravních nákladů. Dopravu je také možné kombinovat (např. zboží z několika kamionů je přesunuto na jeden vlak), čímž dojde k další úspoře nákladů na přepravu. Skladováním sezónních výrobků je možné dosáhnout rovnoměrnější výroby a tím odstranit nepříznivý vliv progresu fixních nákladů na výrobek. Výrazných úspor lze docílit také přesunutím konečné úpravy výrobků do prostoru skladu (např. výstupní kontrola a oprava vad probíhá až v distribučním skladu).

Na zlepšení úrovně služeb mají vliv především mezisklady, které soustřeďují zásoby pro dokončení požadavků dalším článkům distribučního systému. Tím je zabezpečena rychlá reakce na změnu požadovaného sortimentu výrobků. Krátkodobé zásoby sezónního zboží soustředěné krátce před danou sezónou do prodejen nebo dislokovaných skladů zajišťují zrychlení reakce na poptávku a dodací doby. Dalším





specifikem jsou sklady pro drobné podnikatele, které jim umožňují rychlé a méně nákladné pořízení materiálu pro vlastní činnost (ve smyslu velkoobchodního prodeje).

Většina nákladů na skladování má povahu fixních nákladů. Jedná se zejména o:

- Odpisy a náklady na údržbu vybavení skladu
- Náklady na energie, provoz manipulačních prostředků a zabezpečení skladovacích podmínek
- Osobní náklady
- Administrativní náklady
- Obalové a fixační materiály
- Manipulační prostředky (palety, krabice, kontejnery apod.)
- Externí služby

Při výstavbě skladu je tedy nutné dobře zvážit jeho výkonové parametry. K výše jmenovaným nákladům je potřeba připočítat i náklady na ztráty způsobené špatnou manipulací se zbožím ve skladu, nedodržením skladovacích podmínek, překročením záruční doby apod. [5]

## 2.4 Optimalizace ve skladech

Optimalizace ve skladech slouží ke zvýšení jejich výkonnosti a tím přispívá ke snížení nákladů, zvýšení konkurenceschopnosti a zlepšení úrovně služeb zákazníkům. Optimalizace může být provedena ve všech oblastech skladů od jejich technické struktury, přes provozní strukturu až k jejich řízení.

V rámci optimalizace se zaměřujeme na odstranění těchto druhů plýtvání:

- **Nadprodukce** – je důsledkem výroby většího množství výrobků, než zákazník požaduje nebo příliš častými dodávkami zákazníkovi. Její příčinou je zpravidla snaha o dosažení vyššího využití výrobní kapacity a výroba „pro případ



nouze“. Vysoká míra nadprodukce vede k dalšímu zdroji plýtvání, nadbytečným zásobám.

- **Nadbytečné zásoby** – vznikají důsledkem nadprodukce. Jedná se o skladování vysokého počtu materiálu, náhradních dílů, dokončených a nedokončených výrobků atd. To má za následek zvýšení nákladů na skladování a nadměrný kapitál vázaný v zásobách.
- **Defekty** – vznikají na základě produkce nekvalitních a neshodných výrobků. Opravy defektů představují další zvýšení časových i finančních nákladů.
- **Zbytečná manipulace** – je způsobena nevhodným rozvržením pracoviště. Pro pracovníky znamená zbytečné pohyby, např. chození pro různé věci do skříní nebo regálů.
- **Špatné zpracování** – je způsobeno zejména špatným zpracováním výrobního postupu, který může být složitý a časově náročný, což má vliv na zvýšení nákladů procesu.
- **Prostoje** – jsou způsobeny čekáním pracovníků a strojů např. z důvodu nedostatku materiálu, dílů nebo polotovárů. Jejich důsledkem jsou kapacitní ztráty a s nimi spojené náklady, způsobené špatným řízením materiálového toku.
- **Transport** – představuje dlouhé a neefektivní přepravní cesty mezi pracovišti, čímž rostou náklady na přepravu a zvyšuje se riziko poškození výrobků vlivem transportu. [4]

#### 2.4.1 Ukazatele výkonnosti ve skladech

*Ukazatele výkonnosti (angl. Key Performance Indicators – KPI)* se obecně používají k určení účinnosti a efektivity procesu a jsou stěžejní při optimalizaci jakýchkoli procesů, a to i ve skladech. S jejich pomocí lze efektivně monitorovat a vyhodnocovat skladové procesy a tím snížit náklady, odhalit včasné problémy, identifikovat oblasti pro zlepšení a stanovit cíle pro další období. Pro měření výkonnosti se v případě skladů využívá mnoho různých typů KPI.

Například podle [25] jsou 4 oblasti ve kterých lze KPI měřit:

- Spolehlivost – včasné dodání, míra plnění objednávek a přesnost
- Flexibilita – cycle time objednávky
- Náklady – produktivita, poměr nákladů a tržeb
- Využití aktiv – využití skladového prostoru, vytížení pracovníků a strojů

Jiné rozdělení znázorněné v Tab. 1 definuje 25 různých ukazatelů za pomoci skladových aktivit. [10]

Tab. 1 – Přehled ukazatelů výkonnosti ve skladech [10]

	Financial	Productivity	Utilization	Quality	Cycle Time
<b>Receiving</b>	Receiving cost per line	Receipts per man-hour	% Dock door utilization	% Receipts processed accurately	Receipt processing time per receipts
<b>Putaway</b>	Putaway cost per line	Putaways per man-hour	% Utilization of putaway labor and equipment	% Perfect putaways	Putaways cycle time (per putaway)
<b>Storage</b>	Storage space cost per item	Inventory per square foot	% Locations and cube occupied	% Locations without inventory discrepancies	Inventory days on hand
<b>Order picking</b>	Picking cost per order line	Order lines picked per man-hour	% Utilization of picking labor and equipment	% Perfect picking lines	Order picking cycle time (per order)
<b>Shipping</b>	Shipping cost per customer order	Orders prepared for shipment per man-hour	% Utilization of shipping docks	% Perfect shipments	Warehouse order cycle time

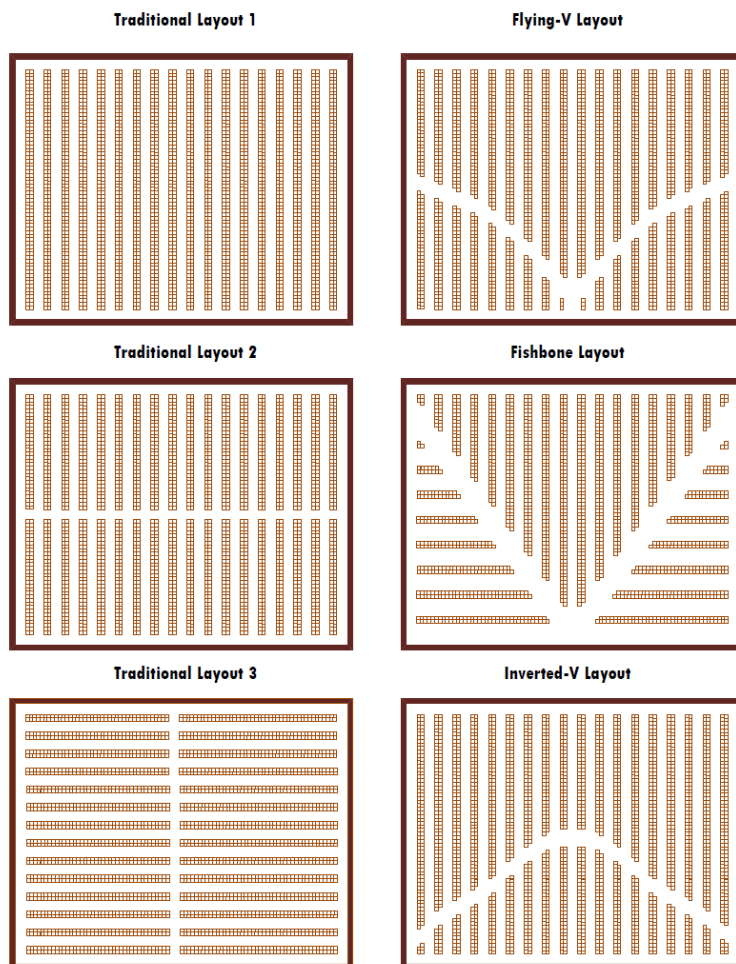
## 2.4.2 Optimalizace technické struktury

Součástí základní technické struktury jsou např. výběr a dimenzování dopravníků a skladových zařízení, návrh rozložení regálů ve skladu, popřípadě celého logistického centra a další elementy, které souvisí s technickou strukturou. Návrh rozvržení skladu je klíčovou součástí při jeho optimalizaci a má značný vliv na délku přepravních vzdáleností a odbavování výrobků ve skladu. Při řešení rozvržení skladu je třeba brát v úvahu následující vlastnosti:

- Počet skladových bloků
- Rozměry a počet uliček skladu
- Počet a tvar křížení uliček skladu
- Počet úrovní skladovacích regálů

- Pozice vstupních a výstupních bran do a ze skladu

Na Obr. 15 jsou znázorněny nejčastější typy rozvržení skladů v rámci řešení jeho



Obr. 15 - Nejčastější typy rozvržení skladů[9]

technické struktury.

Optimalizace technické struktury se dále může zaměřovat na používání standardizovaných nosičů (např. palet, gitterboxů apod.), dopravníků, kontrolování materiálového toku pomocí RFID čipů nebo čárových kódů, případně kompletní automatizaci skladu. [9]

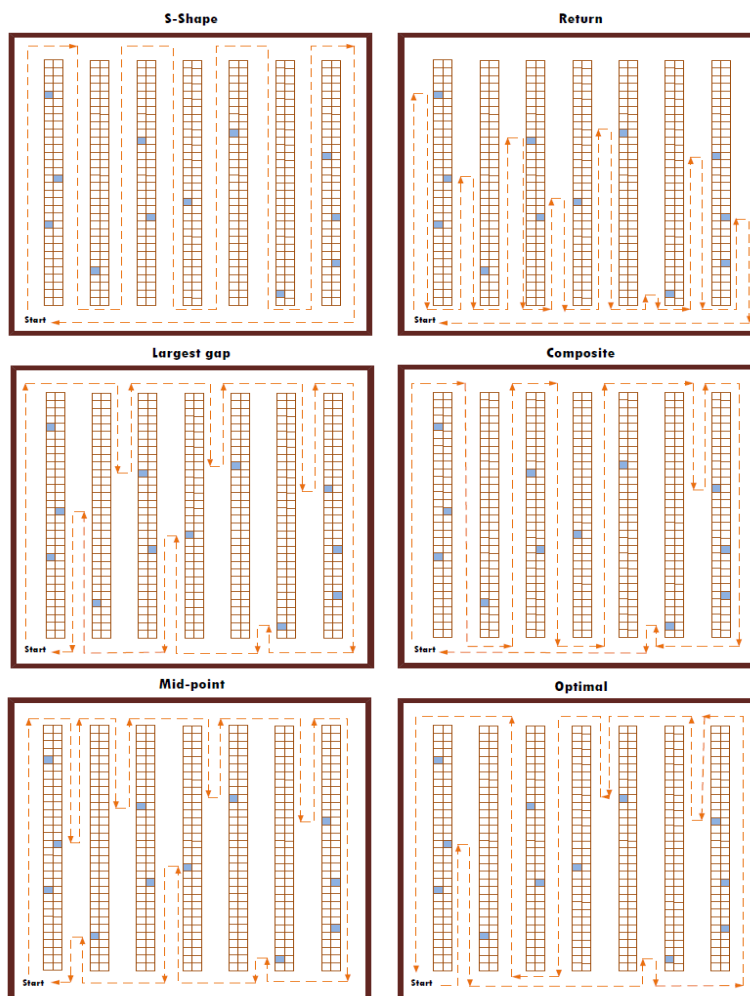


### 2.4.3 Optimalizace provozní struktury

Do provozní struktury řadíme aspekty z několika oblastí např. řízení podniku, řízení zásob, řízení dopravy a další. Tyto aspekty řeší otázku optimalizace umístění položek ve skladu. Jejím cílem je snížit náklady na manipulaci, zkrátit výdejní dobu položek atp. V rámci optimalizace přidělování skladovacích pozic se využívají dvě strategie: náhodná a vyhrazená. Při náhodné strategii je paleta s výrobky uložena na libovolnou volnou pozici nebo na nejbližší prázdnou pozici. Při vyhrazené strategii je paleta uložena pouze na předem určenou pozici (např. podle kódu pozice). Skladovací pozice jsou často organizované, např. při *skladování podle tříd* jsou výrobky skládány na pozice podle četnosti jejich objednávek. Dalším typem organizovaného skladování je *skladování do rodin*, při kterém jsou pozice voleny na základě podobnosti výrobků, či objednávek.

S přidělováním skladovacích pozic výrobkům je úzce spjata i jejich vyskladňování. Jednou z nejpoužívanějších strategií v této oblasti je *vyskladňování jednotlivých objednávek*. S vyskladňováním objednávek je spojena *skladová jednotka* (*angl. Stock Keeping Unit – SKU*). Jedná o nejmenší fyzickou jednotku výrobku, se kterou společnost manipuluje např. paleta, krabice, box aj. Nedílnou součástí procesu vyskladňování jsou také *opatření vyskladňovacích cest* (*angl. Routing policies*). Tato opatření zajišťují

efektivní vyskladňování objednávek po optimální cestě. Na Obr. 16 je zobrazeno šest nejčastěji používaných typů cest.



Obr. 16 - Nejčastěji používané typy vyskladňovacích cest[9]

Další možností, jak optimalizovat provozní strukturu skladu z hlediska vyskladňování je vyskladňování výrobků po dávkách (angl. *Batching*) a rozdělení vyskladňování do zón (angl. *Zoning*). Vyskladňování po dávkách má velký vliv na výkonnost celého procesu vyskladňování, avšak je mnohdy složité, a proto je stále zkoumáno. Vyskladňování do zón je oproti tomu jednodušší, ale s nižším efektem na výkonnost procesu. Jeho nevýhodou je také náročnost složení objednávky, pokud je vyskladňována z různých zón. [9]

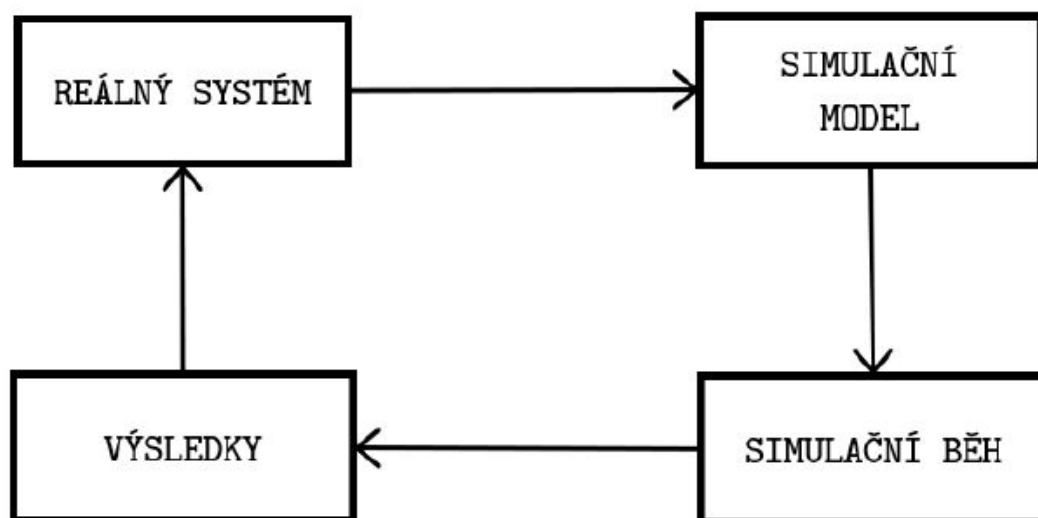
#### 2.4.4 Optimalizace řízení skladu

V rámci řízení skladů mají zvláštní význam systémy koordinace a řízení. Systémy řízení skladů (angl. Warehouse Management Systems – WMS) se používají kromě řízení i k optimalizaci skladových operací. Za jejich pomoci je možné zjistit každý detail o jakémkoliv výrobku, jeho aktuální umístění ve skladu, vytížení skladu, počtu objednávek a pracovníků anebo řídit tok lidí, strojů a výrobků. Na tyto řídicí systémy bývají většinou napojeny ostatní systémy např. informační systémy, plánovací systémy, systémy řízení zdrojů a toku materiálu a další. Optimalizace v této oblasti je důležitá zejména pro zlepšení výkonnosti podniku jako celku a zvýšení jeho konkurenceschopnosti.[9]

#### 2.4.5 Využití simulací při optimalizaci

Simulace jsou využívány při optimalizaci skladování v různých oblastech, jako je výroba, distribuce, maloobchod, a další. Pomocí simulací je možné rychle analyzovat a optimalizovat efektivitu skladování zásob, snížit náklady na skladování a zajistit co největší dostupnost produktů pro zákazníky. Dále je pomocí simulací možné modelovat různé scénáře skladování, optimalizovat rozložení skladů a zásob ve skladových prostorech, optimalizovat procesy naskladňování a expedice zboží, a také plánovat strategie obnovování zásob. Simulace těchto scénářů umožňují manažerům lépe pochopit fungování skladových procesů a tím udělat tyto procesy rychlejší a efektivnější, což má za následek snížení nákladů a zvýšení produktivity. Jiným využitím simulací může být tvorba tzv. digitálního dvojčete tedy virtuální kopie skladu. S jeho pomocí lze sledovat většinu probíhajících procesů v reálném čase, a tak ještě rychleji analyzovat tyto procesy a reagovat na možné změny a problémy. **Chyba! Nenalezen zdroj odkazů.**[11]

## 3 Simulace



*Obr. 17 - Obecný průběh simulace[13]*

Simulace je děj, při kterém se napodobují funkce určitého reálného nebo plánovaného systému, objektu, či procesu. Základním kamenem simulace je simulační model, který vzniká zobrazením a zjednodušením reálného systému. Obecný průběh simulace je zobrazen na Obr. 17.

Simulace jsou důležitým nástrojem při poznávání a zejména při optimalizaci různých systémů a procesů. Jejich uplatnění lze nalézt v mnoha oborech například výroby, logistice, stavebnictví aj. Jednou z nejpoužívanějších metod simulací je počítačová simulace, jíž se především tato kapitola věnuje. V úvodu je rozebrána historie simulace od programovacích jazyků až ke novodobým grafickým simulačním softwarům. Dále jsou ze popsány jednotlivé typy simulací, simulačních softwarů a jejich vlastnosti. Konec kapitoly se zaměřen na využití, přínosy a náklady simulací. [12]

### 3.1 Historie simulací

Simulační nástroje se postupně vyvíjejí již od jejich prvních zmínek v 50. letech 20. století. Přibližně do 80. let se však jedná pouze o programovací jazyky určené pro



simulace. S vývojem se měnila i komplexnost modelů a nároky na simulaci se zvyšovaly. V 90. letech již simulační programy nabízeli uživatelům jednoduché grafické prostředí, čímž se značně zlepšil uživatelský komfort a simulační modely byly přehlednější. Na přelomu tisíciletí se objevují první programy s 3D grafickým prostředím. Tím dochází k dalšímu usnadnění práce se softwarem a zlepšení přehlednosti modelů. V dnešní době dokážou simulační programy spolupracovat s virtuální realitou a nabízejí také různé knihovny a doplňky např. pro optimalizaci modelu.

S rozvojem simulačního softwaru dochází ke snižování doby pro tvorbu modelu a nároků na uživatele. Zatímco v počátku simulací bylo zapotřebí velmi dobrých znalostí programování a dané problematiky, dnešní programy obsahují již předpřipravené objekty s kódem, které uživatel jen přizpůsobí daným požadavkům. I díky tomu se simulace stala dnes velmi vyhledávaným nástrojem pro optimalizaci prakticky jakýchkoliv procesů.[12][13]

## 3.2 Dělení simulací

Simulace jsou velmi široký pojem používaný v mnoha oborech. Z tohoto důvodu existuje vícero kritérií, podle kterých lze simulace rozdělit do skupin. Základní rozdělení je tedy možné udělat podle oboru ve kterém se používají, např.: matematické simulace, biologické a fyzikální simulace, simulace proudění plynů a kapalin anebo průmyslové simulace. Dále lze simulace dělit podle způsobu vytvoření simulačního modelu. Většina simulací je vytvořena na základě matematických modelů. Simulační modely jsou v tomto případě reprezentovány matematickými funkcemi. Jedním ze zástupců vytvořených na tomto základě jsou i počítačové simulace. Jelikož je tato práce zaměřena především na ně, jsou v této kapitole uvedeny a blíže popsány pouze její následující typy: disktrétní, spojitá, deterministická a stochastická.

### 3.2.1 Disktrétní simulace

Hlavní prvky, které prochází disktrétním simulačním systémem, se nazývají entity. Entit je mnoho druhů a představují například objednávky, pracovníky, vybavení,

polotovary a další. Každá entita má poté dané atributy a charakteristiky. Entity se mohou v simulačním modelu také shlukovat do skupin. Tyto shluky se dělí na sety, soubory, listy a řetězce. Cílem disktrétního simulačního modelu je naučit se o dynamickém chování systému za pomoci zobrazení aktivit ve kterých jsou entity zapojeny. Toho je dosaženo vytvořením aktivit, které přesouvají systém z jednoho stavu do druhého. Začátek a konec každé aktivity se nazývá událost. Mezi dvěma po sobě jdoucími událostmi zůstává stav modelu nezměněn, tudíž konstantní. [12]

### 3.2.2 Spojitá simulace

Stav systému ve spojitém simulačním modelu je reprezentován závislými proměnnými, které se mění spojitě v čase. Jelikož tyto proměnné charakterizují stav simulovaného systému, jsou často označovány jako stavové proměnné. Spojitý simulační model je vytvořen pomocí soustavy diferenciálních či integrálních rovnic, které definují tyto proměnné. Typickou nezávislou proměnnou používanou v rovnicích je čas.

Řešení simulace se získá určením a dosažením hodnot stavových proměnných v počátečním (nebo požadovaném) stavu do rovnic a použitím výsledků těchto rovnic jako nových stavových proměnných pro další výpočty. Tento postup tvorby simulačního modelu se nazývá spojitá simulační analýza. Tato analýza se využívá v případě, kdy jsou proměnné popsány explicitní funkcí nebo diferenčními rovnicemi. V případě že jsou rovnice na sobě vzájemně závislé, je řešením simulace numerické řešení soustavy rovnic v každém okamžiku simulace. [12]

### 3.2.3 Deterministická simulace

Simulace se za deterministickou označuje v případě, že se v ní nevyskytují žádné pravděpodobné (náhodné) části. Příkladem může být komplikovaný systém diferenciálních rovnic popisující chemickou reakci. Výstup z deterministického modelu je vyhodnocen poté, co je specifikována množina vstupních veličin a vztahů v modelu, a to i přes to, že je vyhodnocení velmi časově náročné. [14]

### 3.2.4 Stochastická simulace

Stochastické simulační modely pracují s proměnným, které se mění náhodně podle určité pravděpodobnosti. Náhodné proměnné jsou použity do modelu a zaznamenají se jeho výstupy. Poté je tento proces zopakován s novým souborem náhodných proměnných. Tato pakování jsou prováděna několikrát (jedná se řádově desítky až tisíce opakování) po sobě. Výsledkem je distribuce, která zobrazuje nejpravděpodobnější hodnoty a odhady a také jejich očekávané hranice[14][15][14]

### 3.3 Základní vlastnosti simulačních SW

Pro zpracování samotné simulace se v dnešní době využívají specializované softwary. Při výběru simulačního softwaru pro danou aplikaci, je velmi důležité se zaměřit na jeho vlastnosti a správně je vyhodnotit. Tím lze předejít neočekávaným problémům např. s funkčností softwaru nebo jeho komptabilitou. Hodnocené vlastnosti jsou následující:

- **Uživatelská přívětivost** – přehlednost uživatelského prostředí, rychlost odezvy, snadné ovládání softwaru, dostupnost dokumentace, existence různých fór, komunit a helpdesku.
- **Flexibilita modelování** – možnost vytvořit komplexní model, schopnost využít logické a matematické funkce, možnost snadné změny parametrů modelu, možnost uložení a znovupoužití již vytvořených modelů a jejich modifikace.
- **Interoperabilita** – možnost propojení softwaru s dalšími softwary a nástroji prostřednictvím komunikačních protokolů.
- **Možnost týmové spolupráce** – možnost snadného sdílení dat a spolupráce v týmech.
- **Deployment** – způsob zpracování a ukládání dat (lokální výpočetní stanice, cloud, aj.), možnost nastavení datového výstupu (jeho přesnost, formát, vzhled, cílové zařízení apod.).



- **Kvalita výstupů** – přesnost a přehlednost výstupů, možnost modifikace výstupů.
- **Prostorové limity** – možnost provedení simulace bez omezení, v libovolném měřítku.
- **Otevřenost SW** – možnost přímého přidávání dalších funkčních nadstaveb nebo balíčků, možnost tvorby a používání vlastních doplňků.[14]**Chyba!**  
**Nenalezen zdroj odkazů.**

### 3.4 Popis vybraných simulačních SW

Tato kapitola je věnována přehledu vybraných simulačních softwarů. Do výběru byly, z důvodu jejich velmi podobnému zaměření, zařazeny softwary Arena, Plant Simulation, Flexsim a Visual Components. Všechny vyjmenované lze využít k řešení problémů, které jsou podobné problému řešenému v praktické části práce.

#### 3.4.1 Arena

Jedná se o diskretní simulační software od společnosti Rockwell Automation. Umožňuje modelovat, analyzovat a optimalizovat procesy v různých odvětvích, jako je výroba, logistika, zdravotnictví a služby. Vytváření modelu probíhá pomocí bloků (modulů) vývojového diagramu, které zastávají určitý proces, objekt nebo logiku. Bloky jsou na sebe napojeny spojovacími čarami, které představují tok entit systémem, čímž vznikne vlastní simulační model. Simulační model je možné přehledně zobrazit ve 2D i 3D grafice (viz. Obr. 18). Dále tento software nabízí širokou škálu statistických analýz a

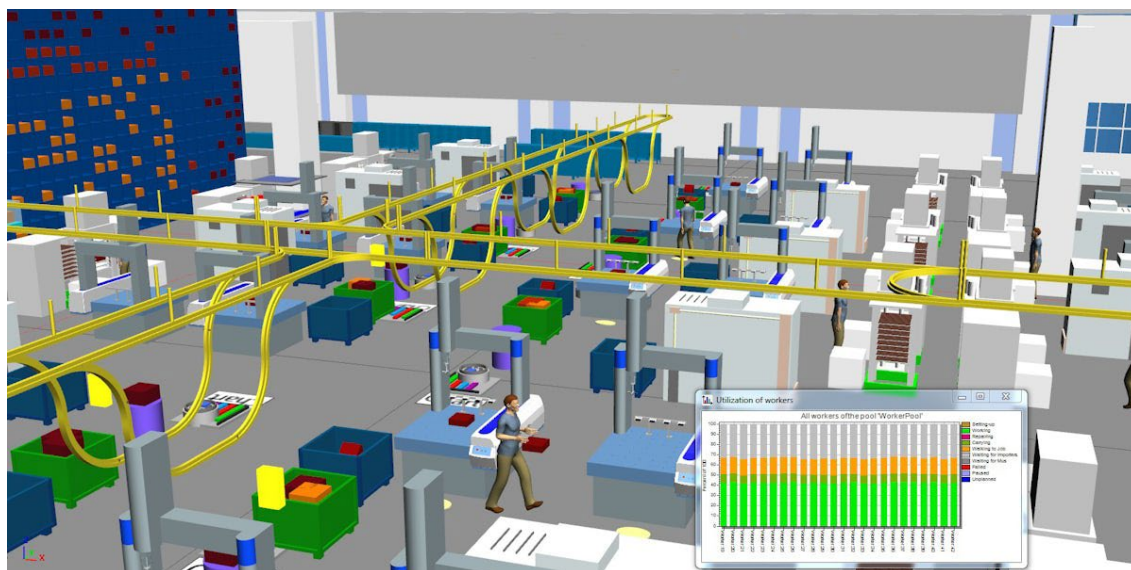


Obr. 18 – Ukázka prostředí softwaru Arena **Chyba! Nenalezen zdroj odkazů.**

optimalizačních nástrojů pro vyhodnocení výkonnosti systému a identifikaci oblastí pro zlepšení. **Chyba! Nenalezen zdroj odkazů.**

### 3.4.2 Plant Simulation

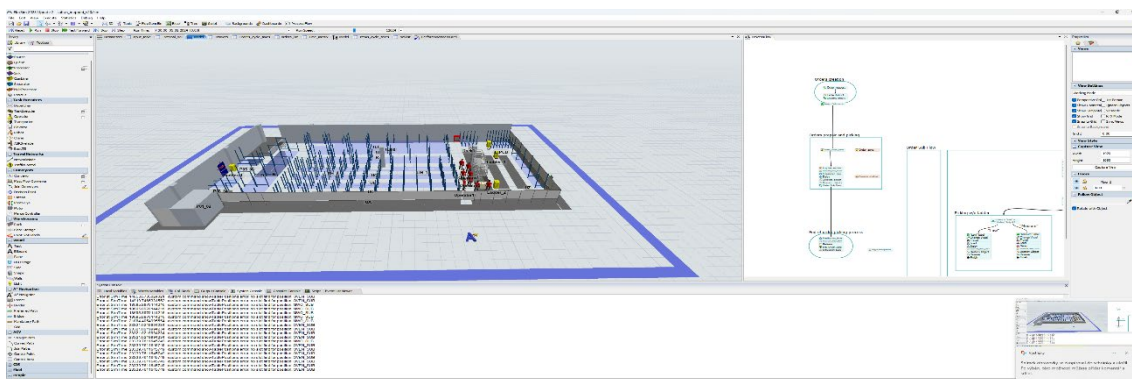
Tento simulační software byl vyvinut společností Siemens a slouží především k tvorbě průmyslových simulací, modelování různých podnikových procesů a rozvrhování pracovišť a podniků. Modelování probíhá ve 2D i 3D grafickém prostředí (viz. Obr. 19). Součástí objektů ve 3D prostředí jsou různé animace jejich pohybů, které přispívají k celkové přehlednosti modelu. Dalším využitím tohoto softwaru je tvorba digitálních dvojčat. K vytvořenému modelu je možné připojit PLC (Programmable Logic Controller – průmyslový počítač) a různé senzory a získávat tak data z provozu v reálném čase. Plant Simulation taktéž poskytuje pokročilé funkce pro analýzu toku materiálu, využití zdrojů, identifikaci úzkých míst a optimalizaci výrobních plánů. [16]



Obr. 19 - Ukázka prostředí Plant Simulation [16]

### 3.4.3 FlexSim

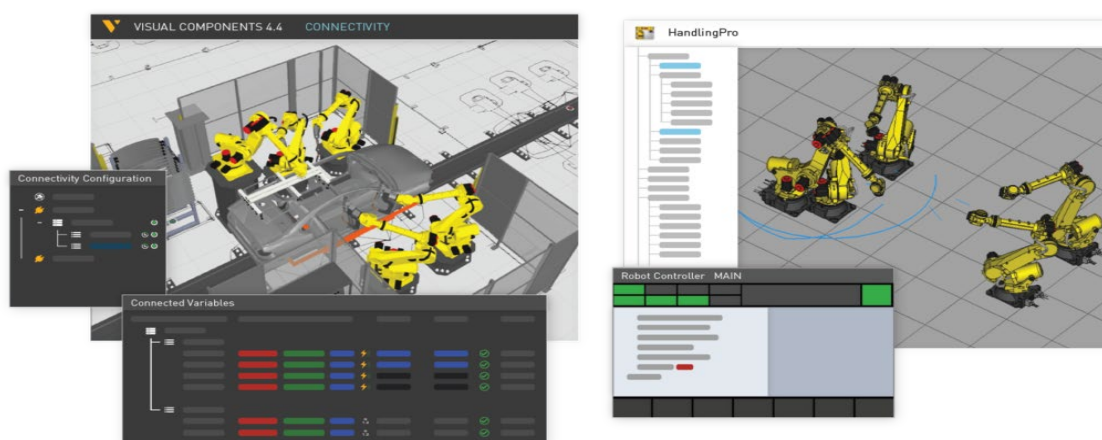
Jedná se o velmi rozšířený simulační software používaný řadou podniků v různých odvětvích jako jsou: výroba, logistika skladování, nebo zdravotnictví. Modelování probíhá, jak ve 3D grafickém prostředí, tak s využitím blokových schémat nazývaných „Process Flow“ (viz. Obr. 20). 3D prostředí je využíváno především pro vytváření realistických modelů, jako jsou např. různé výrobní linky, sklady, či nemocniční zařízení. V Process Flow jsou poté řešeny komplexnější činnosti a procesy v podniku. Obě tato prostředí jsou vzájemně propojena a je tedy např. možné pomocí blokového schématu vytvářet a ovládat různé modely ve 3D prostředí. Výhodou tohoto softwaru je velmi dobrá dostupnost a kvalita naučných materiálů a podpory, jak od vývojářů, tak řady uživatelů.[18]



Obr. 20 - Ukázka prostředí FlexSim

### 3.4.4 Visual Components

V tomto výběru se jedná o jeden z nejnovějších softwarů v oblasti počítačové simulace. Tento nástroj je zaměřený především na simulaci automatizace a robotizace procesů. Uživatelé mohou velmi snadno vytvářet 3D modely robotických buněk, simulovat jejich chování a offline programovat roboty. Podobně jako u všech zmíněných softwarů je i zde k dispozici mnoho knihoven rozšiřujících možnosti použití, možnost napojení na PLC, možnost importu CAD modelů nebo online prezentace a podpora zobrazení ve virtuální realitě. Ukázka prostředí je zobrazena na Obr. 21.[19]



Obr. 21 - Ukázka prostředí Visual Components[19]

## 3.5 Využití simulací

Počítačové simulace mají značné možnosti využití v mnoha oblastech. V dnešní době je možné se s nimi setkat nejen ve výrobě a logistice, ale i ve zdravotnictví, managementu, stavebnictví aj.

Ve výrobě se simulace využívají především analýze faktorů ovlivňujících výrobní procesy. Může se jednat např. o poruchy strojů, délku doby výroby, tok materiálu, nedostatek pracovníků a další. Dále se simulace využívají jako základ pro optimalizaci výrobních procesů a jednotlivých pracovních stanovišť. Zde se jedná např. o odhalení úzkých míst, testování a ověřování nových konceptů, či zvýšení efektivity procesu současně se snížením nákladů. [24]

V logistice jsou simulace využívány především k optimalizaci logistických procesů uvnitř podniku. Dále jsou simulace vhodným nástrojem pro analýzu manipulačních jednotek, toku materiálu a zásob. V rámci optimalizace mají za cíl zvýšení produkce, zrychlení přepravy, zrychlení vyskladnění a naskladnění produktů, snížení počtu pracovníků, snížení nákladů aj. [24]

V managementu je možné simulace využít například k simulování nových návrhů rozložení skladů, továren a kanceláří, využití kapacit strojů a pracovníků, odhadů rizik a cen a simulování obchodních procesů, či ke zkoušení nových podnikových strategií. Také umožňují manažerům studovat chování projektového systému v reálném, zrychleném nebo zpomaleném čase. Již samotné zkušenosti z tvorby modelu mohou vést k návrhům na zlepšení řízení či struktury projektu. Simulační model může sloužit i jako zdroj informací pro operativní plánování kvůli možnosti zahrnutí různých omezení a požadavků.[20][24]

V současnosti je stále více řešeným tématem Průmysl 4.0. Automatizace a robotizace podniků s sebou přináší značné úspory času a nákladů v kombinaci s vysokou flexibilitou. Počáteční náklady na inovaci jsou ale pro podniky velmi vysoké. V tomto případě mohou simulace sloužit jako relativně levný nástroj pro otestování rentability takovéto investice bez zásahu do reálného provozu. Dalším současným trendem jsou tzv. digitální dvojčata. Data z reálného provozu jsou pomocí senzorů přenášena v reálném čase do virtuálního modelu, kde je s nimi možné dále pracovat. Virtuální model se tak poté stává nástrojem pro rychlou analýzu podnikových procesů a umožňuje flexibilně a rychle reagovat na aktuální situace.[12][13]



### 3.6 Přednosti a omezení simulací

V této kapitole jsou popsány výhody a nevýhody použití simulací. Stejně jako ostatní moderní technologie, by se měly použít tehdy, když jejich výhody při řešení daného problému, převažují nad nevýhodami. Je stále třeba brát v potaz, že simulace nejsou jediným nástrojem používaným pro analýzu a existují i jiné levnější a dostupnější varianty. I přes to, že jsou velice flexibilní, jejich použití nemusí nutně znamenat efektivnější a rychlejší zpracování problému oproti jiným metodám. V Tab. 2 je zobrazen přehled nejčastěji řešených problémů a možnosti použití různých metod k jejich řešení. [13]

Tab. 2 - Přehled metod využívaných při návrhu výrobních systémů [13]

Metody Rozbory	Matematické programování	Metody Síťové analýzy	Teorie front/ obslužných sítí	Teorie zásob	Rozmíst'ovací metody	Počítačová simulace
Výrobní program	X					X
Průběžná doba		X	X			X
Stanovení kapacit	X	X	X			X
Využití prostředků	X		X			X
Velikost zásob	X		X	X		X
Dispoziční uspořádání	X	X	X		X	X

#### 3.6.1 Přednosti

Mezi hlavní výhody simulací lze zařadit především:

- Možnost řešení i velmi složitých systémů
- Možnost pozorování chování systému především v reálném čase, ale také ve zrychleném nebo zpomaleném čase
- Možnost pozorování chování systému v dlouhém časovém horizontu
- Simulace nabízejí komplexní pohled na řešený problém a lze zároveň sledovat různé parametry a propojení na další subsystémy
- Větší přehled o zkoumaných procesech za pomoci grafických prvků
- Možnost vzájemného porovnání různých variant řešení



- Snazší a lepší pochopení fungování daného systému
- Rozvoj týmové spolupráce
- Možnost opakovaného použití již jednou vytvořeného modelu, např. v dalších činnostech podniku [13][14][20]

### 3.6.2 Omezení

Mezi hlavní nevýhody simulací lze zařadit především:

- časovou a finanční náročnost
- potřebu speciálního zaškolení
- výstup simulace je závislý na kvalitě a kvantitě vstupních dat
- výstupní informace ze simulace nemusí být i přes jejich vysoké množství užitečné a přesné
- simulace neznamená optimalizace, je to pouze nástroj na pomoc s optimalizací [14][20]

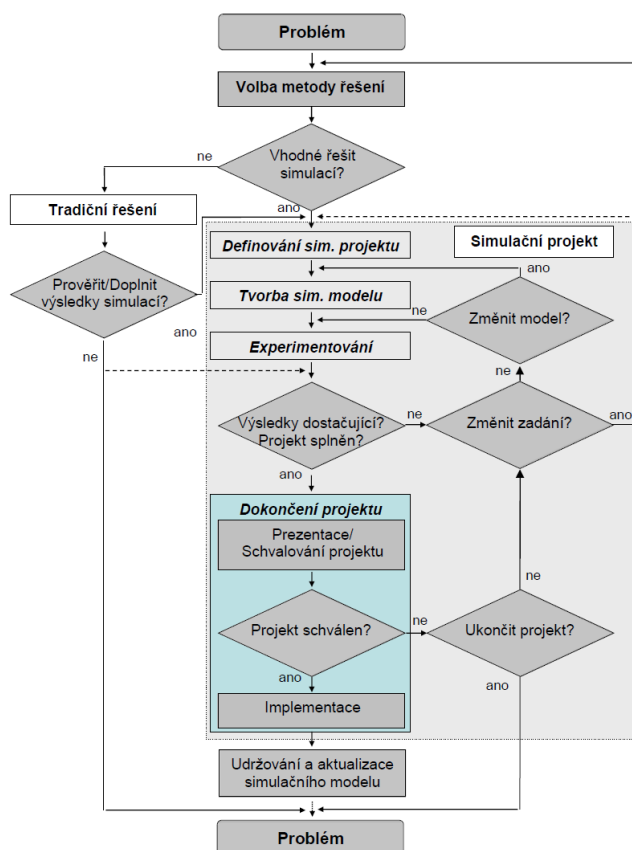
## 4 Simulační projekt

Simulační projekt začíná důkladnou analýzou projektu, definováním jeho cílů a zvolením vhodné metody a postupu řešení. Z těchto výstupů vzniká rozhodnutí, zda má použití simulací pro daný projekt smysl, či nikoliv. Toto počáteční rozhodnutí má také značný vliv na finanční a časové náklady projektu.

Po schválení simulačního projektu je následující postup rozdělen do 4 fází:

- Vymezení projektu
- Tvorba a testování modelu
- Experimentování
- Vyhodnocení simulačního projektu

Schéma tvorby simulačního projektu je zobrazeno na Obr. 22 a jednotlivé části postupu jsou podrobněji popsány dále v této kapitole.[13]



Obr. 22 - Schéma tvorby simulačního projektu [13]

## 4.1 Vymezení projektu

V této fázi se ve spolupráci se zadavatelem studuje, rozebírá a konzultuje daný problém. Po rozhodnutí řešit problém pomocí simulací je vytvořen realizační tým, stanoví se rozsah, cíle a strategie projektu a vytvoří se harmonogram prací. Přestože je tato počáteční etapa nezbytná pro bezproblémový průběh celého projektu, je velmi často podceňována nebo úplně vynechávána. V případě že je tato část kvalitně zpracována, předejde se tak různým nedorozuměním a náhlým změnám v projektu. Tímto lze dosáhnout zkrácení doby trvání tvorby modelu a celého projektu.[13]

## 4.2 Tvorba a testování modelu

Po vymezení projektu a odsouhlasení použití simulace jako hlavního nástroje pro řešení daného problému, se přechází k tvorbě modelu. V této části jsou rozebrány jednotlivé kroky od definice struktury modelu až po jeho finální validaci.

### 4.2.1 Definice struktury

Tvorba modelu začíná definováním jeho struktury. K tomu se využívá tzv. koncepční model. Koncepční model zjednodušeně popisuje plánovaný nebo reálný systém, který se bude simulovat. V této fázi je zapotřebí definovat základní prvky (stroje, pracovníci apod.), vstupy, výstupy, jejich vlastnosti, počet a vazby mezi nimi.

Důležitá je také úroveň podobnosti, ve které se bude model vytvářet. Rozlišují se 2 úrovně podobnosti: izomorfismus a homomorfismus. Izomorfní model má úplnou podobnost s plánovaným/reálným systémem a používá se především pro jednodušší modely. Naopak homomorfní model je zjednodušením plánovaného/reálného systému a je tak vhodný pro větší a složitější modely.[13][22]

#### 4.2.2 Sběr a analýza dat

Tato fáze se zabývá sběrem vstupních dat a jejich analýzou. Kvalita vstupních dat značně ovlivňuje výsledky simulace a tato etapa by rovněž neměla být podceňována. Těmito daty je následně doplňován a zpřesňován koncepční model.

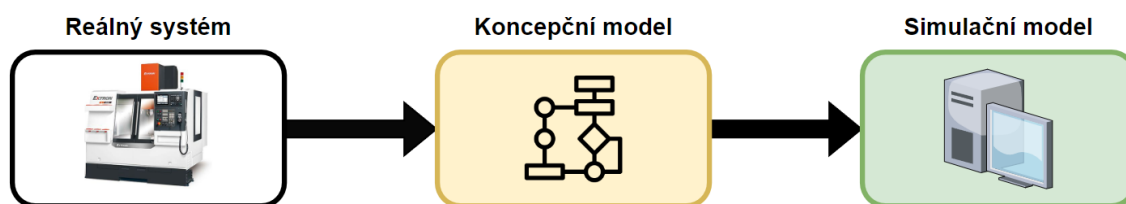
Analýza dat je další důležitou součástí tvorby modelu. Deterministická data (např. rychlost vysokozdvížného vozíku) se dají snadno určit. V praxi má ale většina dat stochastický charakter, tudíž je pro jejich určení potřeba statistická analýza. Tato data jsou tedy obvykle vyjádřena za použití statistických rozdělení. Zde je zobrazen přehled těch nejpoužívanějších:

- **Rovnoměrné rozdělení** – používá se, když jsou zadané informace o problému neúplné, vkládá se minimální a maximální hodnota
- **Trojúhelníkové rozdělení** – používá se, když jsou zadané informace o problému neúplné, vkládá se minimální hodnota, maximální hodnota a modus
- **Binomické rozdělení** – používá se, když je známa pravděpodobnost výskytu daného jevu
- **Poissonovo rozdělení** – používá se, když je znám průměrný výskyt událostí v určitém časovém intervalu
- **Normální rozdělení** – používá se v případech, kde dochází k symetrickým odchylkám od střední hodnoty
- **Erlangovo rozdělení** – používá se, když je znám průměrný časový interval mezi událostmi
- **Weibullovo rozdělení** – používá se zejména pro modelování délky určitého jevu [13]

#### 4.2.3 Tvorba modelu

V této fázi dochází již k samotné tvorbě simulačního modelu v příslušném programu. Tvorba modelu je v podstatě transformace koncepčního modelu do programovacího jazyka, viz Obr. 23. Při transformaci se většinou postupuje od jednodušších modelů ke

složitějším a využívá se řady číselných a grafických prvků, které přispívají k lepší orientaci v modelu. [13][22]



Obr. 23 - Proces tvorby simulačního modelu, vlastní zpracování

#### 4.2.4 Verifikace

Fáze verifikace slouží k ověření přesnosti a funkčnosti vytvořeného simulačního programu, který je porovnáván s konceptním modelem. Při verifikaci se ověřuje správnost:

- logické a obsahové stránky modelu
- převodu modelu do programu
- formální správnost modelu v simulačním jazyce

Model je považován za správný v případě, že jeho chování odpovídá představám konceptního modelu. Některé simulační programy mají v sobě již zakomponované nástroje, které proces verifikace modelu značně usnadňují. Nástroje mohou například kontrolovat správnost zapsání kódu nebo jeho výstupy. [13][12]

#### 4.2.5 Validace

Při validaci dochází k ověření platnosti, zda model správně reprezentuje reálný nebo plánovaný systém. Používané validační techniky se dělí na objektivní a subjektivní. Mezi objektivní techniky se řadí např.: validace vstupů na výstupy a validace s využitím historických dat. Mezi subjektivní techniky se řadí např.: test extrémních podmínek, Turingův test, analýza sensitivity, nebo „face validation“. [12][13]

## 4.3 Experimentování a optimalizace

Ve fázi experimentování dochází k provedení simulačních běhů a zhodnocení jejich výsledků. Na základě výsledků poté případně probíhá optimalizace simulačního modelu.

V první řadě je důležité, aby experimenty respektovaly cíle a požadavky stanovené při vymezení projektu. Aby experimenty poskytly relevantní výsledky je zapotřebí před samotným experimentováním stanovit jednotlivé sledované parametry a jejich hraniční hodnoty kterých mohou dosahovat, účelovou funkci na jejímž základě se budou parametry vyhodnocovat a také počet a délku simulačních běhů.

Dále je vhodné v simulačním běhu zohlednit tzv. zahřívací dobu. Tato doba reprezentuje nestabilní a neustálený stav, při kterém dochází k počátečnímu naplnění zásob do skladů a zařízení a který má zásadní vliv na výsledky simulace a je třeba ho odstranit. Odstranění se provádí např. prodloužením simulačního běhu o zahřívací dobu. Potřebnou délku zahřívací doby je možné určit matematicky, odhadem či graficky.

Každý experiment musí být pečlivě zaznamenán a analyzován z důvodu jeho opakovatelnosti, zjištění odchylek a faktorů ovlivňujících výsledky. Pro získání relevantních výsledků je potřeba každý simulační běh experimentu provést minimálně 3 – 5krát a jeho délka by měla být taková, aby se v něm každá sledovaná událost vyskytla nejméně 10 – 20krát.

Po získání a vyhodnocení výsledků z experimentu se ve většině případů provádí optimalizace simulačního modelu. Optimalizace je proces výběru nejvhodnějšího (optimálního) řešení daného problému. Lze ji provádět tzv. ručně, přičemž se parametry nastavují podle citu a zkušeností daného simulačního inženýra, nebo pomocí optimalizačních metod. Mezi nejčastěji používané optimalizační metody se řadí:

- Enumerace
- Gradientní postupy
- Metaheuristické algoritmy
- Náhodná hledání
- Genetické algoritmy



- Znalostní hledání
- Lineární programování
- Vícekriteriální optimalizace
- A další...[13][23]

#### 4.4 Vyhodnocení projektu

Po provedení všech experimentů a návrhu optimalizace se přechází na poslední část simulačního projektu – vyhodnocení. V této fázi se dokončuje dokumentace k projektu, sepisuje se závěrečná zpráva a provádí prezentace výsledků. V případě potřeby se ještě vylepšuje model po grafické stránce. Po schválení výsledků a samotného projektu nastává fáze zavádění. Zde je zapotřebí aby byl realizační tým důkladně seznámen se simulačním projektem a navrhovanými opatřeními. Ideálním případem je, pokud jsou členové realizačního týmu zároveň tvůrci simulačního projektu.[13]



## 5 Úvod do praktické části

Praktická část této práce je zaměřena na simulaci a následnou optimalizaci rozmístění položek ve výrobním skladu společnosti, vyrábějící interiéry do letadel. Společnost je blíže představena v následující podkapitole. Celý postup simulační studie je proveden podle kroků simulačního projektu popsanych v Kap. 4.

### 5.1 Představení společnosti

Společnost Safran Cabin je součástí francouzské skupiny Safran a specializuje se na výrobu interiérů letadel, jako např. toalety, kabinové skříňky, kávovary, sedadla aj (viz. Obr.24). Jejich 31 závodů zaměstnává okolo 11 000 zaměstnanců po celém světě od Ameriky, přes Evropu, Afriku až po Asii. Obrat celé společnosti se v současné době pohybuje přibližně kolem 25 miliard EUR za rok.



*Obr. 24 - Interiér pro Business jet, Safran Group*

Problém řešený v praktické části této diplomové práce byl řešen pro závod v Thajsku, který spadá pod český závod v Plzni, kde byl společně s jeho zaměstnanci také konzultován. Thajský závod zaměstnává přibližně 700 pracovníků a specializuje se na výrobu elektrických spotřebičů jako jsou kávovary, mikrovlnné a kombinované trouby, chladiče a drtiče odpadků (viz. Obr.25). Skladové prostory v závodě prošly jistou

modernizací, avšak stále jsou manuálně obsluhovány. Jejich hlavním problémem je proces vyskladňování dílů k montáži jednotlivých celků. Díly jsou často vyskladňovány z vysokých pozic za použití žebříku, což značně prodlužuje dobu jejich vyskladnění.



*Obr. 25 - Kombinovaná trouba pro Business jet, Safran Group*

Praktická část této práce řeší vhodné rozmístění dílů (Itemů) ve skladu, pro co nejrychlejší a nejefektivnější vyskladnění objednávek. V rámci simulační studie byla provedena analýza současného stavu, vytvoření 3D modelu pomocí simulačního a modelovacího nástroje FlexSim a dále návrh možné optimalizace.

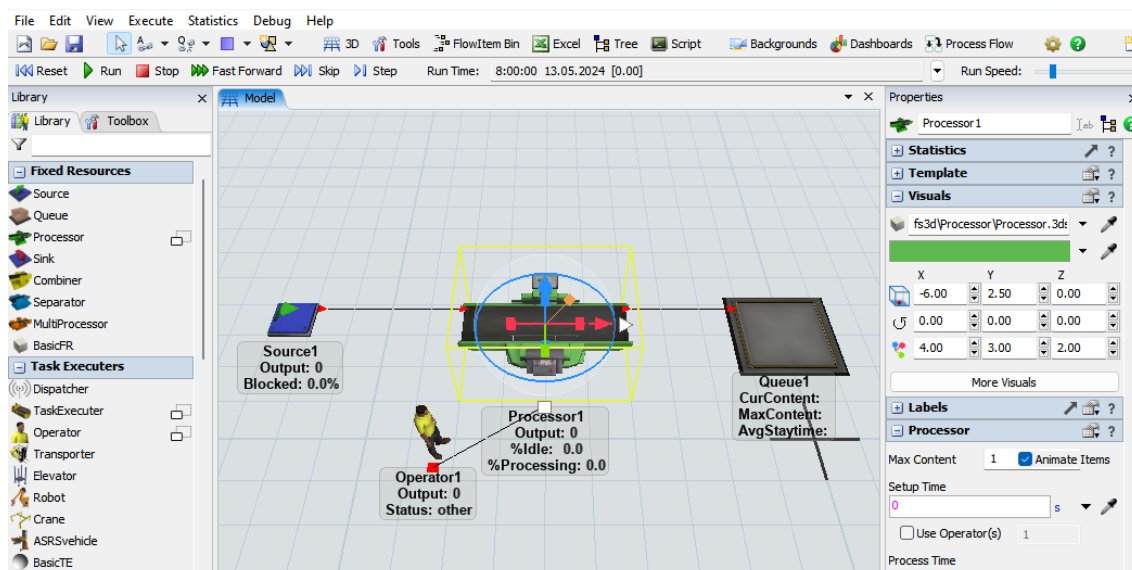
## 6 Popis použitého simulačního nástroje

K realizaci simulace v této práci byl zvolen software FlexSim od společnosti FlexSim Software Products, Inc. Tento software umožňuje uživateli modelovat, simulovat analyzovat a vizualizovat mnoho systémů a procesů v podniku, ať už se jedná o výrobu, logistiku, či skladování. Pro účely této práce byla využita konkrétně verze FlexSim 2023. Zkušební verzi programu lze volně stáhnout po zaregistrování z oficiálních stránek vývojáře: <https://www.flexsim.com/>.

V této kapitole bude vysvětleno uživatelské rozhraní, princip a základní prvky SW použité při tvorbě simulačního modelu a simulace.

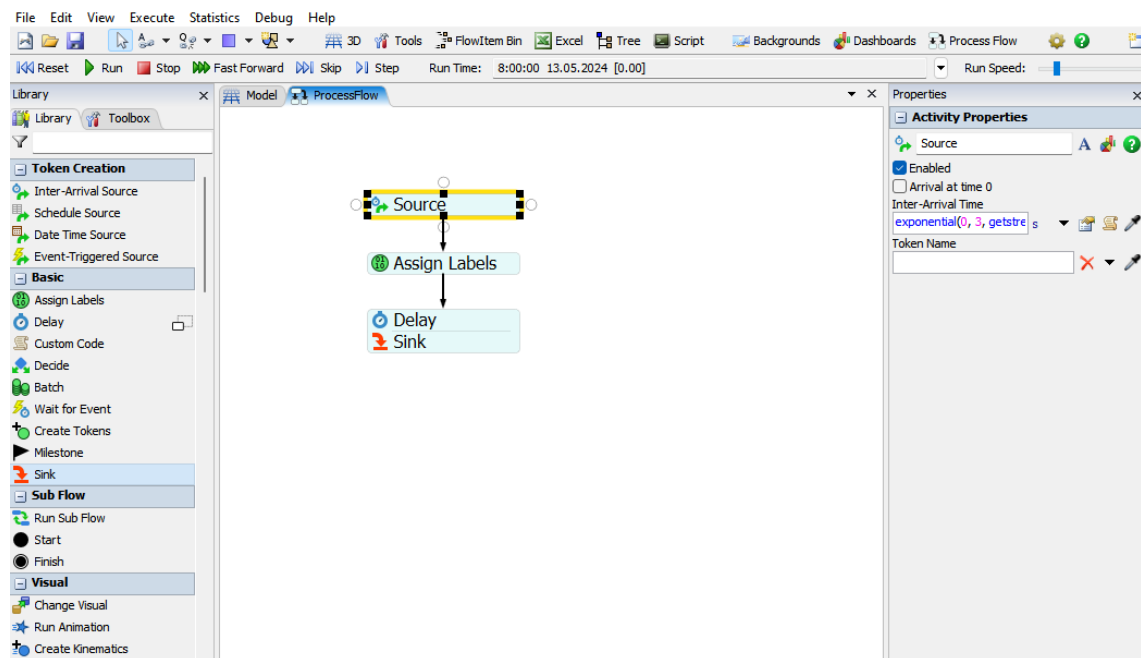
### 6.1 Uživatelské rozhraní

Simulační model je v softwaru Flexsim tvořen především ve 2 základních uživatelských rozhraních: 3D modelu a Process Flow. Prostředí pro tvorbu 3D modelu, zobrazené na Obr., se nachází v záložce Model. Zde uživatel vybírá z knihovny (na Obr. vlevo) jednotlivé objekty a vkládá je na pracovní plochu. Po kliknutí na vybraný objekt se zobrazí okno s jeho vlastnostmi (na Obr. vpravo). Jednotlivé prvky jsou mezi sebou propojeny vazbami.



Obr. 26 - Záložka "Model", vlastní zpracování

Rozhraní Process flow slouží především k vytváření složitějších logik modelu. Z hlediska rozložení jednotlivých funkcí je toto rozhraní velmi podobné tomu pro tvorbu 3D modelu.



Obr. 27 - Prostředí Process Flow, vlastní zpracování

## 6.2 Terminologie

Pro úplné pochopení základů práce se softwarem FlexSim je zapotřebí vysvětlit základní termíny, se kterými tento software pracuje.

### 6.2.1 Objects

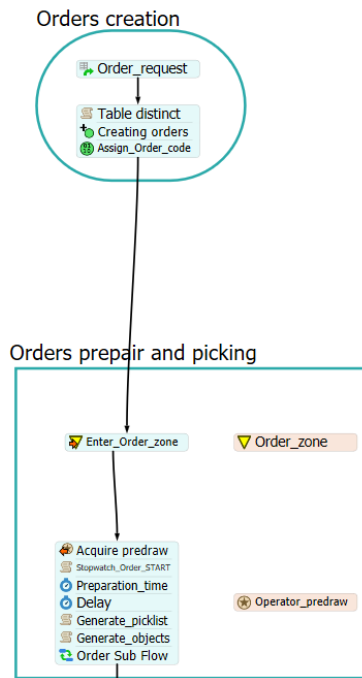
Objects (Objekty) jsou základními stavebními prvky 3D modelu. Ve FlexSimu se nachází několik typů objektů, zde jsou uvedeny ty nejpoužívanější:

- Flow items – objekty které reprezentují materiálový anebo informační tok v modelu. Jedná se například o paletu s výrobky.

- Fixed Resources – statické objekty s danou funkcí. Nejčastěji používanými typy jsou Source, Queue, Processor a Sink. Slouží např. ke generování a odstranění Flow items, nebo práci s nimi.
- Task Executors – pohyblivé objekty vykonávající přidělené aktivity. Jedná se například o pracovníky, vysokozdvizné vozíky, či AGV.
- Warehousing – objekty reprezentující prvky skladů a sloužící k jejich simulaci.
- A\* Navigator – objekty pro navigaci a zamezení vzájemných kolizí mezi Task Executors a mezi Task Executors a ostatními objekty.

## 6.2.2 Process Flow Objects

Process Flow Objects (Objekty procesního toku) (viz. Obr. 28) se používají k modelování aktivit, operací a procesů. S těmito objekty se pracuje ve vlastním modulu Process Flow a nejsou součástí 3D modelu. S pomocí objektů lze vytvořit, jakkoliv složitý proces. Objekty prochází entita zvaná token (podobně jako flow item u Objects). Tokeny při průchodu objektem spouští jeho funkce a tím je možné řídit, jakkoliv složitý proces.



Obr. 28 - Ukázka objektů Process Flow, vlastní zpracování

### 6.2.3 Labels

Labels (Štítky) uchovávají informace o objektu, ke kterému byly přiřazeny (např. typ výrobku, barva, kód výrobku apod.)

### 6.2.4 Lists

Lists (Listy) jsou seznamy položek, které slouží jako databáze pro uchovávání informací o objektech (např. seznam výrobků ve skladu).

### 6.2.5 Tables

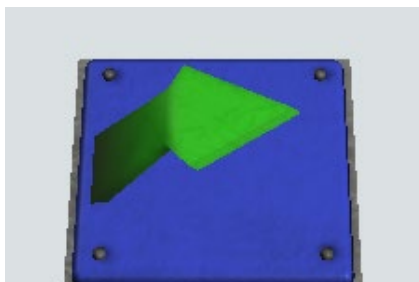
Tables (Tabulky) mají dvojí využití. Buď jsou použity jako zdroj informací pro procesy (např. generování různých typů výrobků podle tabulky), nebo jsou určeny pro uchování informací o daném procesu (např. počet vyrobených kusů). S uloženými informacemi lze v modelu dále pracovat.

## 6.3 Základní objekty 3D modelu

V této podkapitole jsou uvedeny objekty 3D modelu použité při tvorbě simulačního modelu.

### 6.3.1 Source

Objekt „Source“ (Obr. 29) se používá ke generování flow items. Lze u něho nastavit typ generovaného flow itemu a za jakých podmínek dojde k jeho vytvoření (např. počet

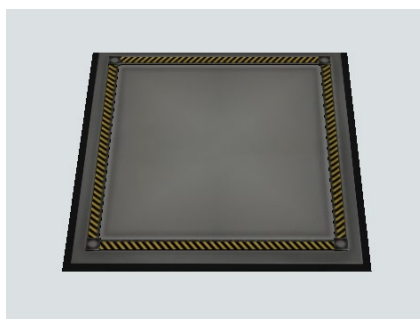


Obr. 29 - Objekt „Source“ [18]

ks za určitou dobu). Vytváření flow itemů probíhá ve vybraném čase, který lze nastavit v mnoha podobách, např. jako pevná hodnota, generace podle statistického rozdělení, případně dle vytvořené tabulky importované ve formátu .XLS(X), nebo .CSV.

### 6.3.2 Queue

Objekt „Queue“ (Obr. 30) představuje mezisklad, místo, kde dochází k dočasnému ukládání příchozích flow itemů (např. prostor na paletu s polotovary, umístěný u obráběcího stroje). Je u něj možné nastavit jeho velikost a typ strategie výdeje flowitemů (FIFO, LIFO atd.).



Obr. 30 - Objekt „Queue“[18]

### 6.3.3 Task Executer

Objekt „Task Executer“ (Obr. 31) se používá k manipulaci s flow itemy. Základními nastavitelnými parametry jsou kapacita, rychlost pohybu, zrychlení, rychlost nakládání a rychlost vykládání. Pro tyto objekty je možné nastavit dráhy po kterých se mají pohybovat. Zároveň lze po jejich přiřazení k „A\* navigatoru“, zamezit kolizím mezi nimi a ostatními přiřazenými objekty.



Obr. 31 - Objekt „Task Executer“[18]

### 6.3.4 Operator

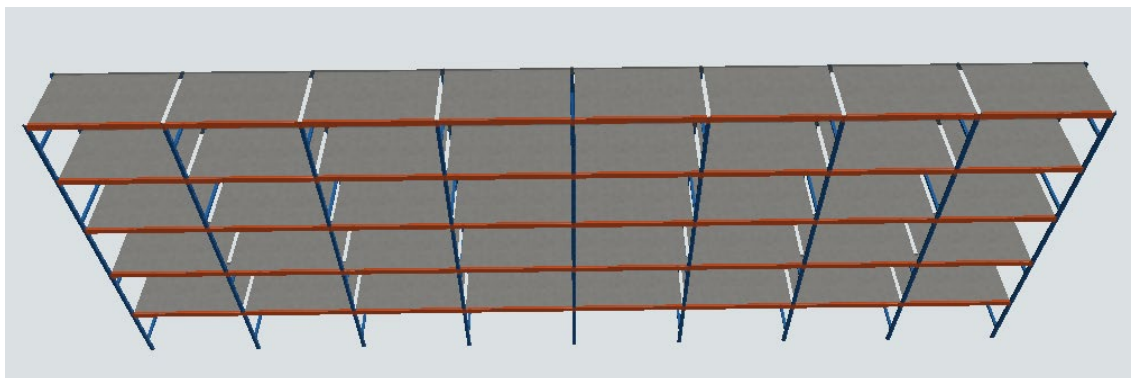
Objekt „Operator“ (Obr. 32) je zvláštním typem objektu „Task Executer“, představující lidského pracovníka, přičemž se využívají kinematiky pohybů specifické pro člověka. Lze u něj nastavit stejné parametry jako u „Task Executeru“. Dále je také možné nastavit několik vizualizací pracovníka.



Obr. 32 - Objekt „Operator“[18]

### 6.3.5 Rack

Objekt „Rack“ (Obr. 33) představuje regálový systém a používá se ke skladování flow itemů. Hlavními parametry, které lze nastavit jsou jeho rozměry, počet řad a sloupců, počet regálů a slotů v regálech a také strategie naskladňování. Na výběr je i široké spektrum vizualizací, které umožňuje regály přesně vizualizovat podle potřeby.



Obr. 33 - Objekt „Rack“[18]



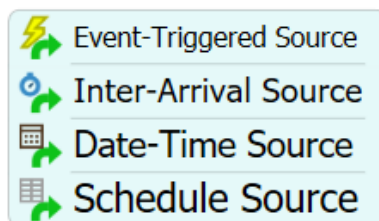
## 6.4 Základní objekty Process flow

V této podkapitole jsou uvedeny objekty Process flow použité při tvorbě simulačního modelu.

### 6.4.1 Source

Objekt „Source“ je v rámci process flow určený k vytváření tokenů. Používají se 4 typy objektu „Source“ (Obr. 34):

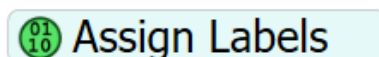
- Event-Triggered Source: tokeny jsou generovány na základě nějaké události v modelu (např. výstup flowitemu z processoru).
- Inter-Arrival Source: tokeny jsou generovány na základě zadaného časového vstupu, který může být například pevně zadaný nebo podle statistického rozdělení.
- Date-Time Source: tokeny jsou generovány na základě zadaného datumu a času.
- Schedule Source: tokeny jsou generovány dle tabulky



Obr. 34 - Typy objektu „Source“ [18]

### 6.4.2 Assign Labels

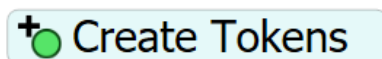
Objekt „Assign Labels“ (Obr. 35) se používá k vytvoření a přiřazení labelu (štítku) k procházejícímu tokenu.



Obr. 35 - Objekt „Assign Labels“ [18]

### 6.4.3 Create Tokens

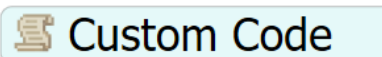
Objekt „Create Tokens“ (Obr. 36) se používá pro vytvoření dalších tokenů, k jemuž dochází při průchodu tokenu objektem. Vytvořené tokeny mohou být typu „Child“, „Siblings“, nebo „Independent“ Tokens.



Obr. 36 - Objekt „Create Tokens“[18]

### 6.4.4 Custom Code

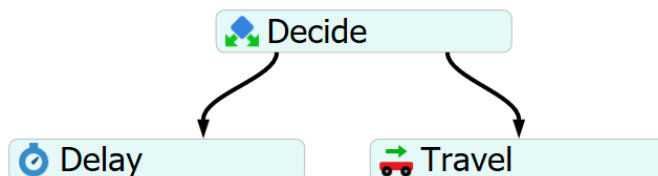
Objekt „Custom Code“ (Obr. 37) umožňuje pomocí programovacího jazyka flexscript, vytvořit libovolný kód. Také je možnost použít již předvytvořené kódy.



Obr. 37 - Objekt „Custom Code“[18]

### 6.4.5 Decide

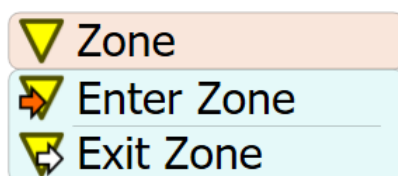
Objekt „Decide“ (Obr. 38) se používá pro vytvoření rozhodovací podmínky. Rozhodování může probíhat na základě mnoha podmínek, např. dle času, podmínky hodnoty, procent, statistického rozdělení atd.



Obr. 38 - Objekt „Decide“[18]

### 6.4.6 Zone

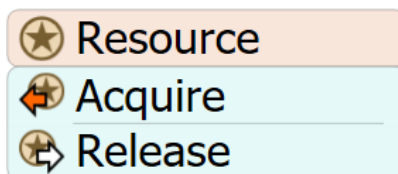
Objekty „Zone“ (Obr. 39) se používají pro definování oblastí v modelu, které slouží např. ke shromažďování statistických informací, přidávání omezení do modelu nebo vykonávání specifických operací pouze v rámci dané zóny. Token vstupuje do zóny průchodem přes objekt Enter Zone. Po vykonání akcí v zóně opět vystupuje průchodem přes Exit Zone.



Obr. 39 - Objekty „Zone“ [18]

### 6.4.7 Resource

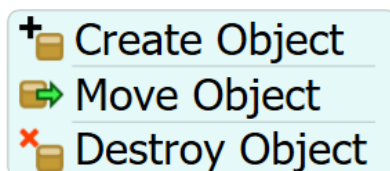
Objekty „Resource“ (Obr. 40) se používají k reprezentaci omezených zdrojů jako jsou např. pracovníci, nářadí, materiál apod. Objekt Resource je získáván pomocí aktivity Acquire Resource a uvolňován pomocí aktivity Release Resource. Tímto způsobem lze efektivně řídit a simulovat využití omezených zdrojů.



Obr. 40 - Objekty „Resource“ [18]

### 6.4.8 Objects

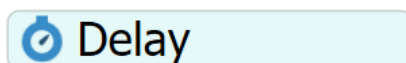
Objekty „Objects“ (Obr. 41) se používají k manipulaci s objekty a daty ve 3D modelu. 3D objekty je možné vytvářet (Create Object), manipulovat s nimi (Move Object), nebo je odstraňovat (Destroy Object).



Obr. 41 - Objekty „Objects“[18]

### 6.4.9 Delay

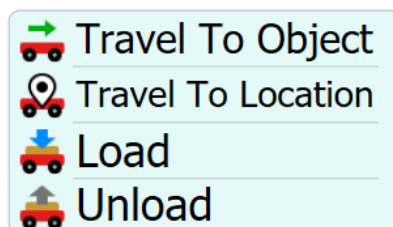
Objekt „Delay“ (Obr. 42) se používá pro časové zpoždění tokenu před jeho dalším průchodem skrz následující objekty Process Flow.



Obr. 42 - Objekt „Delay“[18]

### 6.4.10 Task Sequences

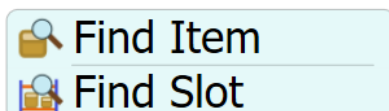
Objekty „Task Sequences“ jsou 3 typů: Travel, Load a Unload (Obr. 43). Objekt typu Travel umožňuje Task Executeru pohyb na pozici určenou souřadnicemi (Travel To Location), nebo polohou objektu (Travel To Object). Objekty typu Load a Unload zajišťují proces nakládky a vykládky daného Task Executera.



Obr. 43 - Objekty „Task Sequences“[18]

### 6.4.11 Warehousing

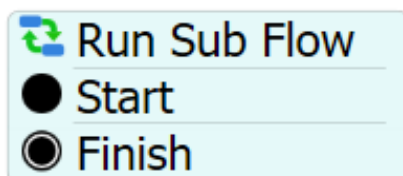
Objekty „Warehousing“ jsou 2 typů: Find Item a Find Slot (Obr. 44). Objekt Find Item se používá k nalezení objektu naskladněného v 3D objektu typu storage (např. Rack). Objekt Find Slot pak přiřazuje danému objektu určitou pozici (slot) ve skladu. Podobně jako u Shared Assets, tak i zde lze objekt či slot hledat na základě dotazu psaného v jazyce SQL.



Obr. 44 - Objekty „Warehousing“[18]

### 6.4.12 Sub – flow

Objekty „Sub – flow“ (Obr. 45) se používají pro zahájení subprocessu. Vstupující token zůstane v objektu, dokud všechny jeho child tokens nedokončí aktivity v subprocessu (projdou objektem Finish).



Obr. 45 - Objekty „Sub-flow“[18]

## 7 Vymezení projektu

Tento simulační projekt se zaměřuje na optimalizaci uspořádání položek ve výrobním skladu jednoho ze závodů společnosti Safran Cabin, za účelem zvýšení efektivity skladových procesů.

V rámci projektu jsou řešeny 2 problémy: analýza současného stavu skladu, zejména počet picků (vzetí dílů) z jednotlivých skladových pozic a optimalizace uspořádání položek ve skladu.

Měření a porovnání výsledků současného stavu je provedeno za pomoci ukazatelů výkonnosti (metrik) definovaných v Tab. 3 Tyto ukazatele byly záměrně zvoleny na základě požadavků společnosti.

Tab. 3 - Metriky simulačního projektu

Metriky simulačního projektu		
Pořadí	Metrika	Jednotka
1	Počet picků z jednotlivých pozic	ks
2	Celkový čas dokončení zkušebního setu objednávek	h
3	Průměrný čas vyskladnění dílu z pozice	s
4	Počet dílů vzatých z neergonomických pozic	ks
5	Využití pracovníků	%
6	Počet zpracovaných objednávek za 1 týden	-

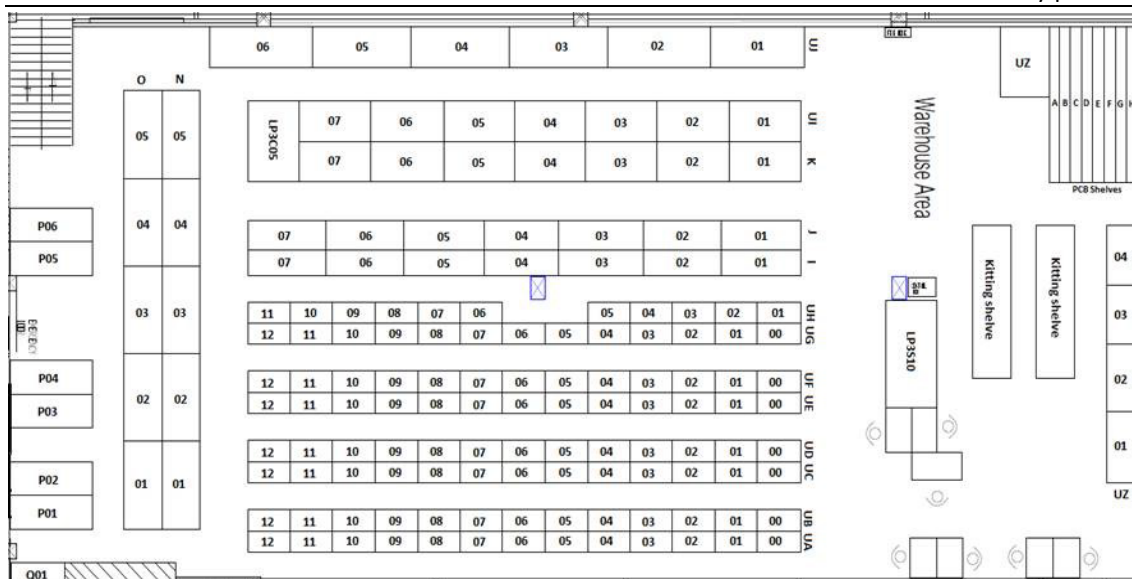
## 8 Koncepční model a sběr dat

Tato kapitola je zaměřena na tvorbu koncepčního modelu, definici datové struktury a sběr dat, potřebných pro vytvoření modelu blížícímu se reálnému systému. V podkapitolách jsou postupně popsány jednotlivé kroky od rozvržení skladu, přes definici simulovaného procesu, až ke sběru dat potřebných pro tvorbu modelu a následnou práci s nimi.

### 8.1 Rozvržení skladu

Ve skladu dílů se nachází soubor několika regálů různých velikostí (viz. Tab.). Jejich rozmístění je zobrazeno na Obr.46. Jednotlivé díly jsou v regálech skladovány v boxech. V rámci skladu se rozlišuje x velikostí regálů a 9 typů úložných boxů (viz. Tab.). Všechny boxy typu „Drawer“ a „Bluebox“ jsou přístupné bez použití žebříku. Boxy typu „Cardbox“ a „Tube“ jsou povětšinou přístupné pouze za použití žebříku.

Ukázka použitých regálů	
Označení regálu	Typ boxu
EIPCB	Drawer (M)
I, J	Blue box (M), Cardbox (L)
K, UI	Blue box (M), Cardbox (L)
O, N	Blue box (M), Cardbox (L)



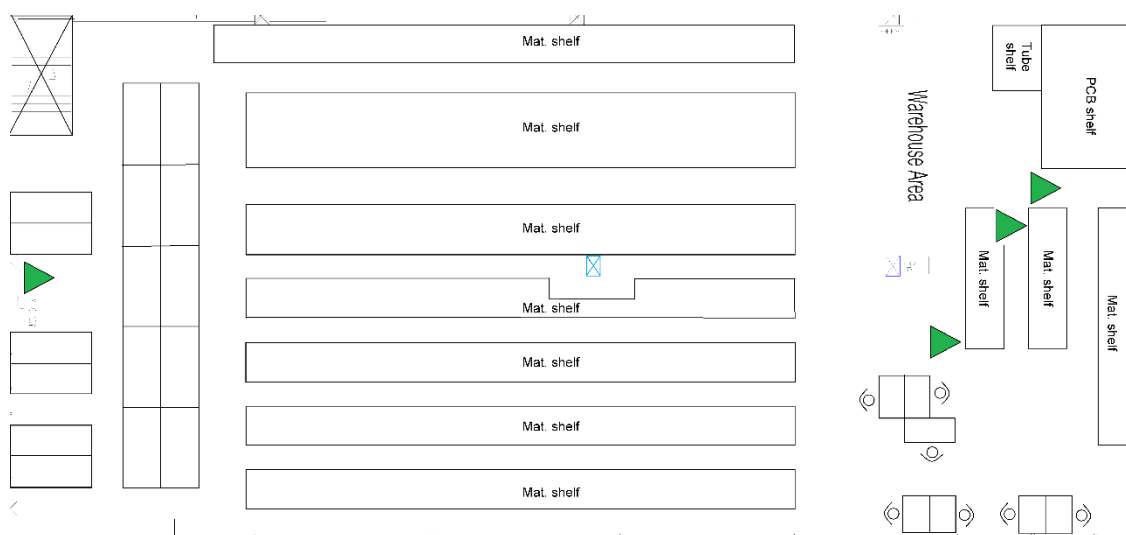
Obr. 46 - - Schéma rozmístění regálů ve skladu

Přehled použitých typů úložných boxů	
Typ úložného boxu	Rozměry (DxŠxV) [cm]
Drawer (XS)	8x12x10
Drawer (S)	22x12x5
Drawer (M)	29x18,8x9,3
Drawer (L)	34,5x26x8
Blue box (S)	50x32,5x17
Blue box (M)	61x42x31
Cardbox (L)	Více rozměrů
Tube (S)	Více rozměrů
Tube (L)	Více rozměrů



## 8.2 Definice procesu

Řešený proces začíná přijetím požadavku (objednávky) na vyskladnění dílů. Po přijetí objednávky provede nejprve přípravu – vezme vozík, vyplní dokumentaci a nahraje objednávku do čtecího zařízení. Po přípravě přechází k vyskladnění objednávky. Seznam dílů v objednávce se nazývá „picklist“. Pracovník skladu má celý picklist pro jednu objednávku nahraný ve čtecím zařízení, se kterým obchází sklad. Jednotlivé díly jsou v regálech rozděleny do několika typů úložných boxů od malých krabiček pro drobné součásti až po rozměrné kartonové krabice a jsou vyskladňovány v abecedním pořadí. Pracovník vyhledá pozici boxu a potvrdí vyskladnění daného dílu načtením jeho kódu čtečkou. Následně vyndá díl z boxu, případně plastového sáčku uvnitř boxu, umístí díl na vozík, box zpět do regálu a pokračuje k pozici dalšího dílu. Většina dílů se nachází v dosahu pracovníka, některé jsou ale dostupné pouze za použití žebříku. Žebříky jsou umístěny na 4 pozicích (viz. zelené trojúhelníky na Obr. 47). Po vyskladnění všech dílů odveze pracovník vozík s díly na výchozí pozici a provede konečnou administrativu (kontrola picklistu, podpis ...). Poté si vozík přebere další pracovník, který ho dopraví do výrobního úseku.



Obr. 47 - Schéma umístění žebříku ve skladu, vlastní zpracování

## 8.3 Sběr dat

Data použitá k tvorbě simulačního modelu byla převzata z reálného provozu. Důležité hodnoty potřebné pro tvorbu modelu jsou:

- Pozice a rozměry regálů ve skladu (viz. Kap. 8.1)
- Pozice boxů v regálech
- Počet picků boxů z jednotlivých pozic
- Složení jednotlivých objednávek
- Pozice žebříků (viz. Obr. 47)
- Časy aktivit v procesu vyskladnění boxů pro jednotlivé velikosti a výšky boxů

Datové soubory s informacemi o pozicích boxů, počtu jejich picků a složení objednávek, bylo zapotřebí upravit za použití MS Excel. Úpravy v konkrétních souborech jsou rozebrány dále.

Data v souboru použitém pro simulaci počtu picků byla pouze pro lepší přehlednost vzestupně seřazena podle kódu pozice (viz. Obr. 48)

1	Locations	Picks
2	EIPCBA01	9
3	EIPCBA02	13
4	EIPCBA03	5
5	EIPCBA04	57
6	EIPCBA05	11
7	EIPCBA06	4
8	EIPCBB01	27
9	EIPCBB02	9
10	EIPCBB03	6

Obr. 48 - Ukázka seřazení dat – simulace počtu picků, vlastní zpracování

Datový soubor pro simulaci procesu vyskladňování obsahoval informace o 575 náhodně vybraných objednávkách (číslo objednávky, pozice dílu, číslo dílu a počet potřebných kusů). Ukázka ze souboru je zobrazena na Obr. 49. Jeho úpravy spočívali v seřazení objednávek vzestupně podle času vyskladnění jednotlivých dílů.

1	Order	Due date	Item	Location	Sum of Qty
2	MLE226590	01.10.2021		82206 UL0101	1
3	MLE275753	15.12.2023	800-2590-01	UC0302C02	1
4	MLE275753	15.12.2023	800-1942-01	UD0108C01	1
5	MLE284401	15.12.2023	800-1942-01	UD0108C01	1
6	MLE281408	29.01.2024	38011-1	UC1202C01	-1
7	MLE285397	16.02.2024	800-2567-01	I06001	0
8	MLE291582	16.02.2024	800-3293-01	J04003	1
9	MLE291583	16.02.2024	800-3293-01	J04003	1
10	MLE291584	16.02.2024	800-3293-01	J04003	1

Obr. 50 - Ukázka seřazení dat – simulace procesu vyskladňování, vlastní zpracování

Rychlosti a doby pohybu pracovníků jsou již součástí simulačního softwaru FlexSim, případně byly dopočteny pomocí MOST analýzy. Časy jednotlivých aktivit ve vyskladňovacím procesu (viz. Obr. 50) byly dopočteny na základě rozboru videa z reálného procesu v kombinaci s MOST analýzou. Výsledné hodnoty byly řádně prodiskutovány a schváleny zaměstnanci společnosti.

Constants		Drawer (XS)		Drawer (M)	
Scan of barcode	2	Pull drawer	1	Pull drawer	1
Counting parts		Open plastic bags	4	Open plastic bag	2
Typing Qty in scanner	3	Pick part	2	Pick part	3
Find position (analysis)	5	Close plastic bags	4	Close plastic bag	4
Put the part in the trolley	2	Push drawer	2	Push drawer	2
Note down the Qty on paper	6	Counting parts	5	Counting parts	3
Compare lot number	10	Label scan+print+	15		
Put the part in the plastic bag	2				

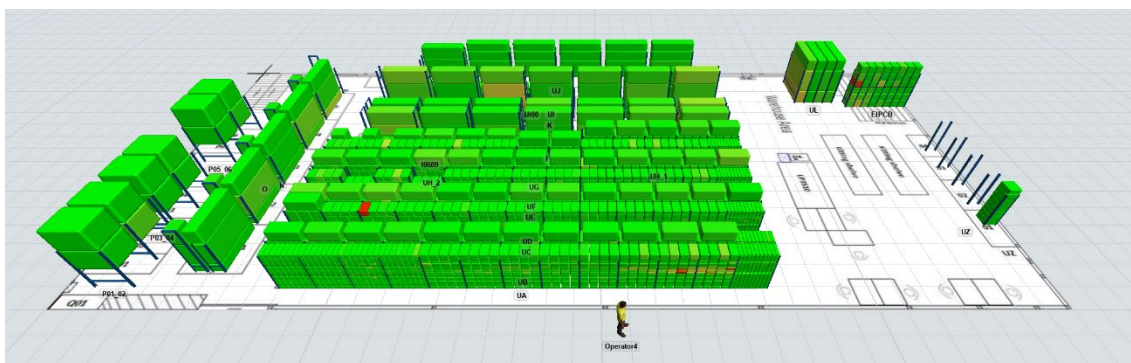
Obr. 49 - Ukázka dat – aktivity v procesu vyskladňování, vlastní zpracování

## 9 Tvorba simulačního modelu

Podle koncepčního modelu byly následně vytvořeny 2 simulační modely. První model slouží především k informativním účelům a má za úkol vizualizovat počty picků z jednotlivých regálů. Tento model je tak základem pro pozdější optimalizaci polohy boxů. Druhý model je již komplexnější. Jeho hlavní úlohou je simulace pohybu pracovníků skladu na jejímž základě je možné provést optimalizaci skladu. Dále v této práci je rozebrán první model ve zkrácené formě a druhý model v podrobné formě.

### 9.1 Model č.1 – 3D model a Process flow

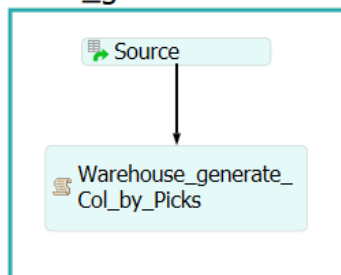
3D model prvního simulačního modelu (viz. Obr. 51) je tvořen pouze regály reprezentovanými objekty typu Rack a rozmístěnými podle layoutu reálného systému. Vstupní data jsou nahrána v tabulce *Pivot\_Safran*. V té jsou vypsány jednotlivé pozice dílů a ke každé z nich počet picků za vybrané měsíce (dohromady 38dní). Podle těchto pozic jsou v objektech Rack namapovány jednotlivé sloty (skladové pozice) (viz. Kap.9.2.1). V modelu jsou použity celkem 4 typy mapovacích schémat.



Obr. 51 - Model č.1 - 3D model, vlastní zpracování

Logiku modelu má na starosti blok *Item\_generation* (viz.Obr. 52) v *Process flow*, který zajišťuje naplnění regálů boxy s pomyslnými díly tzv. flowitemy a jejich obarvení podle počtu picků. Při spuštění simulace dojde k vygenerování tokenu, který následně vstoupí do objektu *Warehouse\_generate\_Col\_by\_Picks* (Custom Code) (viz Obr. 53).

### Items\_generation



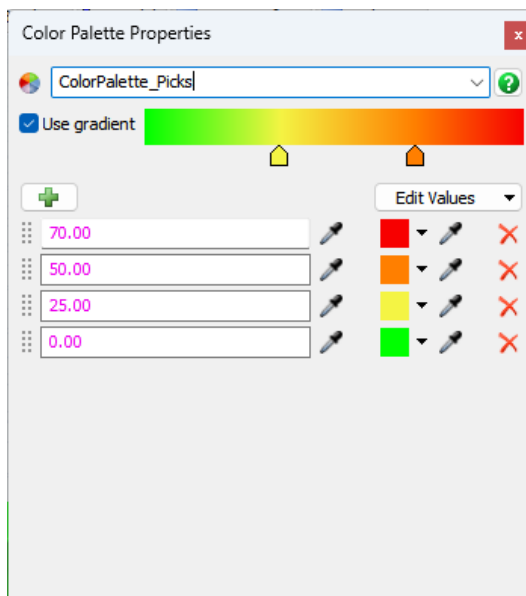
Obr. 53 – Blok „Items\_generation“, vlastní zpracování

```

1 /**Custom Code*/
2 Object current = param(1);
3 treenode activity = param(2);
4 Token token = param(3);
5 treenode processFlow = ownerobject(activity);
6
7 Table table = Table("Pivot_Safran");
8 treenode flowitem = Model.find("MODEL:/Tools/FlowItemBin/drawer/drawer");
9 createTableFlowitems_2(table.as(treenode), 0, flowitem);
  
```

Obr. 52 Programový kód objektu Warehouse\_generate\_Col\_by\_Picks, vlastní zpracování

U všech regálů ve 3D modelu je nastavený trigger – On Slot Entry, který přidá flowitemům při vstupu label slot (reprezentující pozici v regále) a nastaví jim barvu podle počtu picků. Barvy jsou nastavovány v barevné škále (viz. Obr. 54), díky čemuž má uživatel velmi dobrý přehled o tom, které pozice jsou nejvíce využívány. Hodnoty u jednotlivých barev jsou upravitelné podle aktuálních potřeb uživatele. Pro potřeby simulace v této práci, jsou barvy nastaveny tak, že každá barva odpovídá čtvrtině z nejvyššího počtu picků.



Obr. 54 - Barevná škála pro počet picků, vlastní zpracování

## 9.2 Model č.2 – 3D model

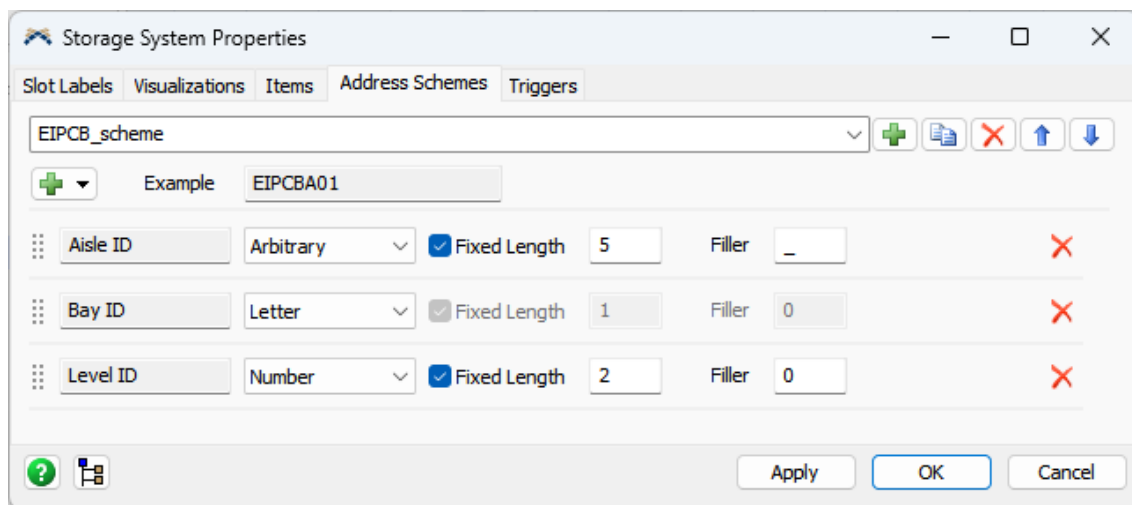
Základem 3D modelu druhého simulačního modelu je layout vytvořený v Google SketchUp a importovaný do 3D modelu FlexSimu. Do tohoto layoutu jsou dále umístěny 3D objekty. 3D model je složen pouze ze skladu (skupina objektů Rack), pracovníků (Operator) a objektu A\* Navigator. Logika fungování celého procesu ve skladu, tedy propojení skladu a pracovníků je následně vytvořena za použití Process flow (viz. Kap. 9.3).

### 9.2.1 Sklad

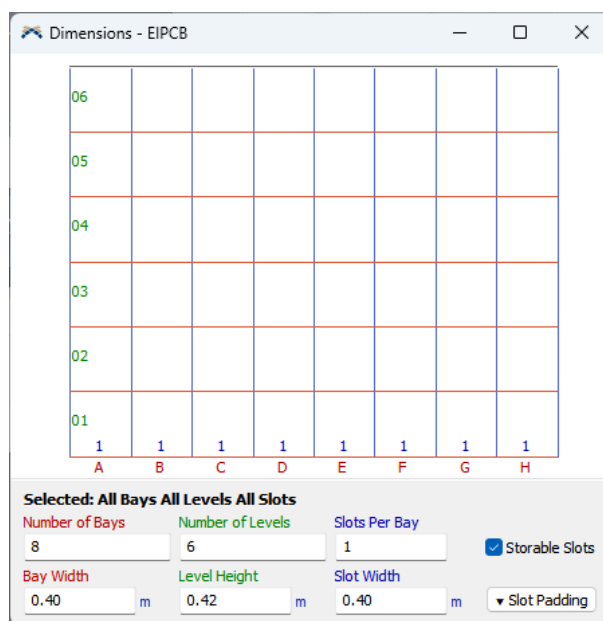
Sklad je složen z několika objektů typu Rack, reprezentujících regály. Velikosti a polohy jednotlivých objektů jsou stejné jako v reálném systému (viz. Kap. 8.1). Adresy jednotlivých slotů (pozic) v regálech jsou namapované podle kódů pozic jednotlivých dílů podle kódů pozic. Ukázka mapování pozic je zobrazena na Obr. 55 a Obr. 56. V modelu jsou použity celkem 4 typy mapovacích schémat.

Generování (zaskladnění) boxů do Racků je zajištěno pomocí objektu Generate\_objects (Custom Code) z bloku Individual picklist generation v Process flow

(viz. Kap. 9.3). Boxy jsou vygenerovány vždy pouze do pozic podle konkrétní objednávky, která je zpracovávána. Toto řešení je zvoleno z důvodu nižšího vytížení hardwaru při simulaci. Vyskladnění boxů má na starosti pracovník – Operator.



Obr. 56 - Jeden ze 4 typů mapovacích schémat použitých v modelu, vlastní zpracování

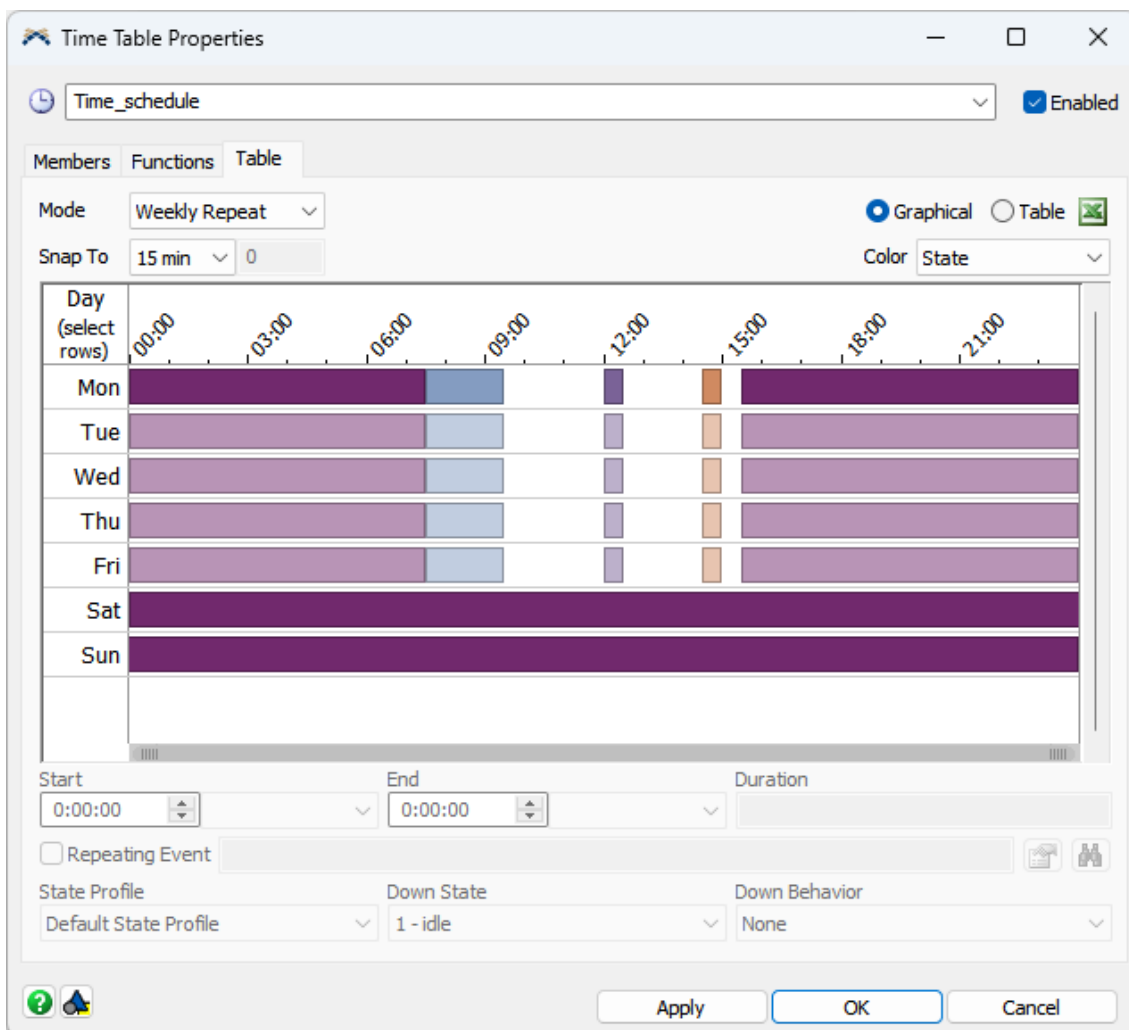


Obr. 55 - Ukázka rozmístění skladových pozic (slotů) regálu EIPCB, vlastní zpracování

## 9.2.2 Pracovníci

Pracovníci jsou objekty typu Operator. Jejich hlavním úkolem je simulovat pracovníky reálného skladu, především jejich chování při procesu vyskladňování objednávek. Modeloví pracovníci vykonávají práci pouze po dobu stanovenou časovým

rozvrhem, který je vytvořen podle reálného systému (viz. Obr. 57). Barevně jsou označeny doby, kdy pracovníci nevykonávají žádnou práci. Pracovníci pracují 5 dní v týdnu od 7:30 do 15:30. V rámci pracovní doby probíhají vždy od 7:30 do 9:30 meetingy, od 12:00 do 12:30 oběd a od 14:30 do 15:00 pracovní pauza.



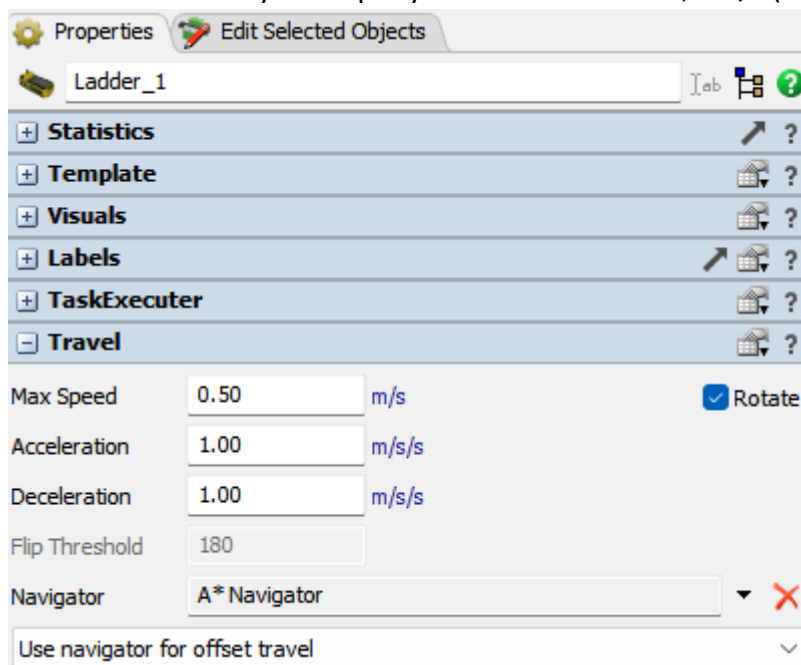
Obr. 57 - Časový rozvrh pracovníků: tmavě fialová – čas mimo pracovní dobu, světle modrá – ranní meetingy, světle fialová – obědová pauza, oranžová, pracovní pauza, vlastní zpracování

### 9.2.3 Žebříky

Ve skladu se nachází také 5 žebříků – Ladder. Ty slouží pracovníkům k sundání boxů umístěných ve výšce označené „High“. Každý žebřík má svoji pozici a pracovník si vždy



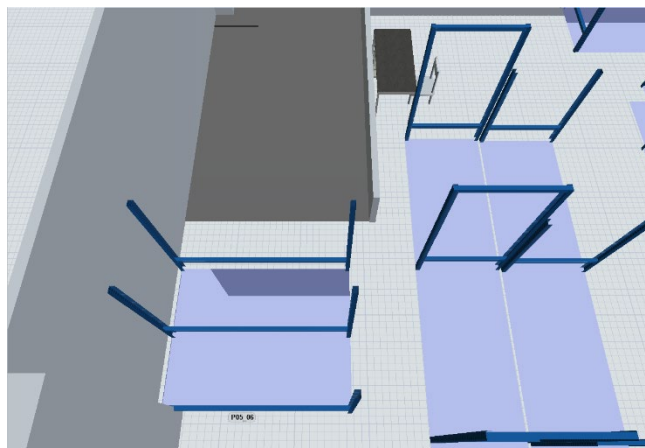
volí ten který je k němu v danou chvíli nejbližší. Po použití vrací pracovník žebřík zpět na jeho původní pozici a pokračuje dále ve vyskladňování. Při pohybu pracovníka se žebříkem v reálném systému dojde ke snížení jeho rychlosti pohybu. Z tohoto důvodu je i v simulačním modelu snížena rychlost pohybu se žebříkem na 0,5 m/s (viz. Obr. 58).



Obr. 58 - Rychlost pohybu se žebříkem, vlastní zpracování

#### 9.2.4 A\* Navigator

Tento objekt zde slouží pouze pro vytvoření pomyslných bariér, které zabraňují pohybu pracovníků a žebříků do určitých míst, nebo skrze 3D objekty. Bariéry jsou vytvořeny např. v okolí regálů, stolů, schodiště, nebo na místě pomocných regálů (viz. Obr.59).

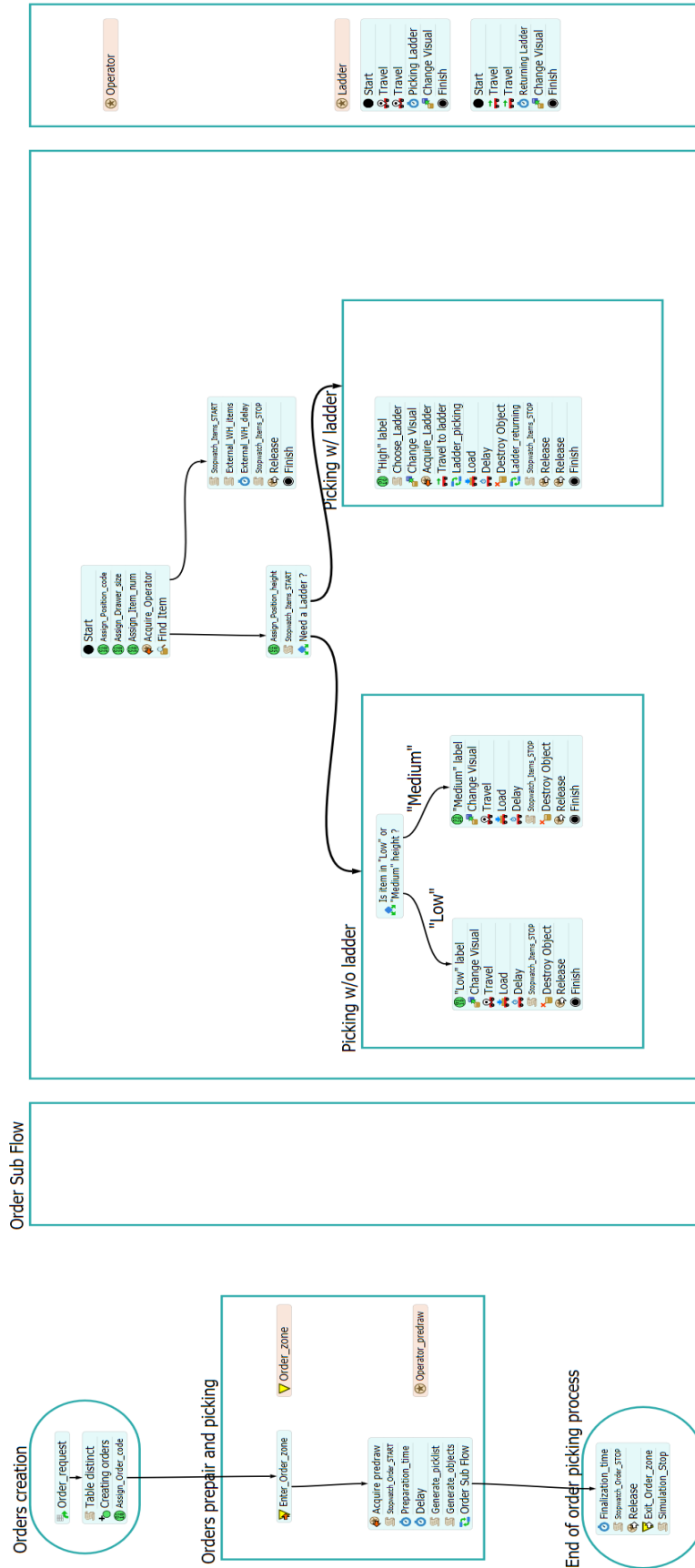


Obr. 59 - Ukázka bariér objektu A\* Navigator: šedá – schodiště, světle modrá – prostor regálů, vlastní zpracování



## 9.3 Model č.2 – model Process Flow

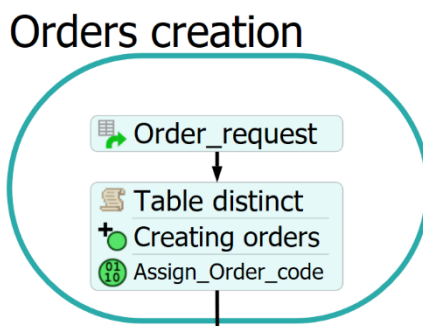
Ke správnému pochopení fungování tohoto modelu je důležité porozumět termínům popsaným v Kap.6. Model Process flow pro druhý model (viz. Obr. 60) byl vytvořen na základě vývojového diagramu zobrazeného v Kap. 8.2. Model je pro lepší přehlednost rozdělen do 4 bloků: Orders creation (Vytvoření objednávek), Orders prepair and picking (Příprava a pickování objednávek), Order Sub Flow a End of order picking process (Konec pickovacího procesu). Jednotlivé objekty Process flow jsou pojmenovány a v závorce je uvedeno, o jaký typ objektu v knihovně se jedná, např.: Order\_request (Source).



Obr. 60 - Přehled jednotlivých bloků Process Flow pro 2. model. vlastní

### 9.3.1 Orders creation

Tento blok (Obr. 61) je začátkem celého procesu vyskladnění objednávek. Objekt `Order_request` (Source) vytvoří počáteční token, který zastupuje požadavek na vyskladnění objednávek.



Obr. 61 - Blok `Orders_creation`, vlastní zpracování

Vytvořený token následně vstupuje do objektu `Table_distinct` (Custom Code), kde spustí sestavený programový kód. Zde prvně dochází k načtení vstupní tabulky `input_table` z excelového souboru (viz. Obr. 63). Tento soubor je generován podnikovým systémem a obsahuje informace o složení objednávek a pozice jednotlivých dílů. Po načtení se z tabulky vyfiltrují pouze kódy objednávek, které jsou následně uloženy do nové tabulky `orders_list` (viz Obr. 64).

```

1 /**Custom Code*/
2 Object current = param(1);
3 treenode activity = param(2);
4 Token token = param(3);
5 treenode processFlow = ownerobject(activity);
6
7 Table distinct = Table("orders_list");
8 Table orders = Table("input_table");
9 Table result = Table.query("SELECT Order FROM input_table GROUP BY Order");
10 result.cloneTo(Table("orders_list"));
11

```

Obr. 62 – Programový kód objektu `Table_distinct`, vlastní zpracování

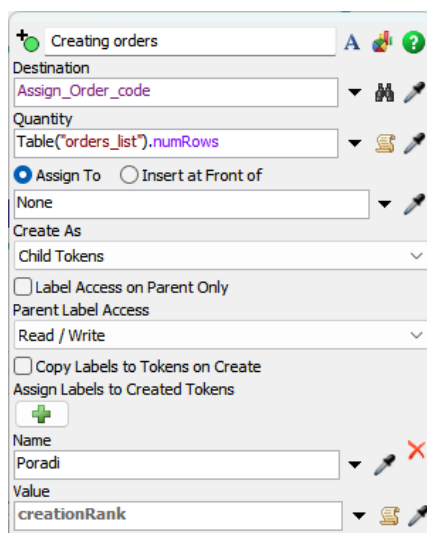
	Order	Due date	Item	Location	Sum of Qty
	MLE226590	44470	82206	UL0101	1
	MLE275753	45275	800-2590-01	UC0302C02	1
	MLE275753	45275	800-1942-01	UD0108C01	1
	MLE284401	45275	800-1942-01	UD0108C01	1
	MLE281408	45320	38011-1	UC1202C01	-1
	MLE285397	45338	800-2567-01	I06001	0

Obr. 65 – Vstupní tabulka input\_table

	Order
1	MLE226590
2	MLE275753
3	MLE284401
4	MLE281408
5	MLE285397
6	MLE291582
7	MLE291583

Obr. 64 – Tabulka orders\_list, vlastní zpracování

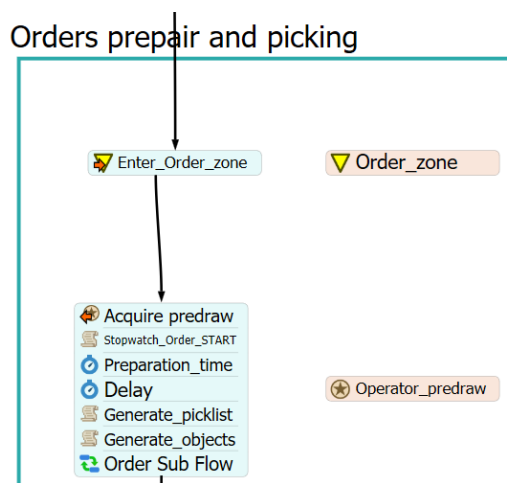
Dalším krokem je vytvoření počtu tokenů podle počtu objednávek, každý token tak následně odpovídá jedné objednávce. To má na starosti objekt Creating\_orders (Create Tokens). Tento objekt také přiřadí tokenům label Poradi, který přidává tokenu informaci o poradí jeho vzniku. Dále je každému tokenu objektem Assign\_Order\_code (Assign label) přiřazen label s unikátním kódem objednávky. Nyní jsou vygenerovány

Obr. 63 - Nastavení objektu Creating\_orders,  
vlastní zpracování

všechny objednávky a přechází se ke generování picklistů a následně procesu jejich vyskladnění.

### 9.3.2 Orders prepair and picking

V tomto bloku (Obr. 66) se vytvářejí picklisty pro jednotlivé objednávky, naskladňují se objekty do virtuálního skladu a simuluje se vyskladnění objednávek. Tokeny vygenerované objektem *Creating\_orders* (viz. Kap. 9.2.1) zde nejprve vstoupí do *Order\_zone* (Zone). Tento objekt zajišťuje, aby v jeden moment byla zpracovávána pouze jedna objednávka (nedojde k vpuštění dalšího tokenu – objednávky do dané zóny).



Obr. 66 - Blok *Orders prepair and picking*, vlastní zpracování

Po vstupu tokenu do *Order\_zone*, je každému tokenu objektem *Acquire\_predraw* (Resource *Acquire*) přiřazen 1 z 8 pracovníků, který je v daný moment volný (nepracuje na žádné objednávce). Pokud není volný ani jeden pracovník, čeká token, dokud se některý neuvolní. Dále je token poslán do objektu *Stopwatch\_Order\_START* (Custom Code), který při jeho průchodu zaznamená aktuální čas simulace (viz. Obr.67). Tento čas je následně zapsán do tabulky *Orders\_cycle\_times* (sloupec *Time IN*) a reprezentuje čas přijetí objednávky (viz. Obr. 68). Pro vyšší přehlednost a usnadnění práce s daty, jsou informace o jednotlivých objednávkách zapsány na nový řádek.

```
1 /**Custom Code*/
2 Object current = param(1);
3 treenode activity = param(2);
4 Token token = param(3);
5 treenode processFlow = ownerobject(activity);
6
7 Table table = Table("Orders_cycle_times");
8 table.addRow();
9 token.time_table_ref = table.numRows;
10
11 table[table.numRows][1] = token.Order_code;
12 table[table.numRows][2] = Model.time; //tady uloz nazev te pozice
13
```

Obr. 67 – Programový kód objektu Stopwatch\_Order\_START, vlastní zpracování

Order	Time IN [s]	Time OUT [s]	Cycle_time [min]
MLE226590	7200	7440.80	4.01
MLE275753	7200	7493.78	4.90
MLE284401	7200	7445.49	4.09
MLE281408	7200	7439.91	4.00
MLE285397	7200	7450.53	4.18
MLE291582	7200	7496.20	4.94

Obr. 68 - Tabulka Orders\_cycle\_times, vlastní zpracování

Dále proces pokračuje přípravou podkladů. Zde je tento úkon simulován objektem Preparation\_time (Delay). Jeho výchozí hodnota je nastavena na 60 sekund. Poté je na základě dat z tabulky *input\_table*, objektem Generate\_picklist (Custom Code) vygenerována tabulka *picklist* (viz. Obr. 69). Ta je generována vždy pro jeden token (objednávku) a obsahuje informace o jednotlivých dílech v objednávce (číslo dílu, jeho poloha ve skladu a velikost boxu v jakém je díl umístěn) (viz. Obr. 70). Tabulka *picklist* je vstupem pro objekt Generate\_objects (Custom Code), který vytváří boxy na příslušných pozicích regálů, resp. v objektech Rack, které je ve 3D modelu zastupují (viz. Obr.71).

V tento okamžik jsou připravena všechna potřebná data a přechází se k procesu vyskladnění objednávek.

```
1 /**Custom Code*/
2 Object current = param(1);
3 treenode activity = param(2);
4 Token token = param(3);
5 treenode processFlow = ownerobject(activity);
6
7 Table orders = Table("input_table");
8 Table drawers = Table("Drawers");
9 Table result = Table.query("SELECT Location, Item, Order, Box_type \
10 FROM Drawers \
11 INNER JOIN input_table ON Drawers.Location = input_table.Location\
12 WHERE Order = $1 ORDER BY Location ASC", token.Order_code);
13
14 result.cloneTo(Table("picklist"));
```

Obr. 71 - Programový kód objektu *Generate\_picklist*, vlastní zpracování

	Location	Item	Order	Box_type
1	EIPCBA03	136-1621	MLE287378	Drawer (M)
2	EIPCBC02	136-1624	MLE287378	Drawer (M)
3	EIPCBC02	136-1622	MLE287378	Drawer (M)
4	EIPCBD04	136-1619	MLE287378	Drawer (M)
5	EIPCBD04	136-1620	MLE287378	Drawer (M)
6	EIPCBF04	136-1618	MLE287378	Drawer (M)
7	UA0110C01	MS35338-136	MLE287378	Drawer (S)
8	UA0111C01	MS35338-136	MLE287378	Drawer (S)
9	UA0112C02	AN960-C10	MLE287378	Drawer (S)
10	UA0210C03	MS51957-30	MLE287378	Drawer (S)

Obr. 70 - Tabulka *picklist*, vlastní zpracování

```
1 /**Custom Code*/
2 Object current = param(1);
3 treenode activity = param(2);
4 Token token = param(3);
5 treenode processFlow = ownerobject(activity);
6
7 Table table = Table("picklist");
8 treenode flowitem_ref = Model.find("/Tools/FlowItemBin/drawer/drawer");
9 createTableFlowitems_2(table.as(treenode), 0, flowitem_ref);
```

Obr. 69 - Programový kód objektu *Generate\_objects*, vlastní zpracování

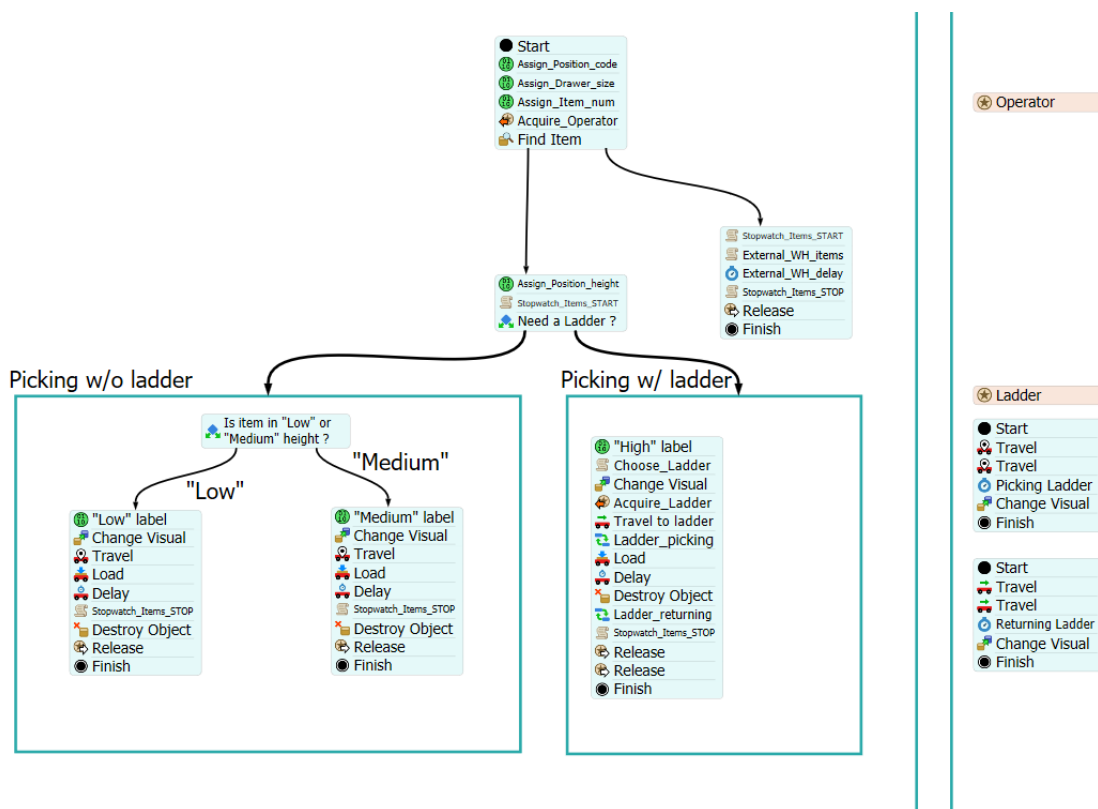
### 9.3.3 Order Sub Flow

Po přípravě podkladů, vygenerování picklistů a 3D objektů je spuštěn subproces Order Sub flow (Sub flow), který simuluje samotné vyskladnění objednávek (viz. Obr 72).

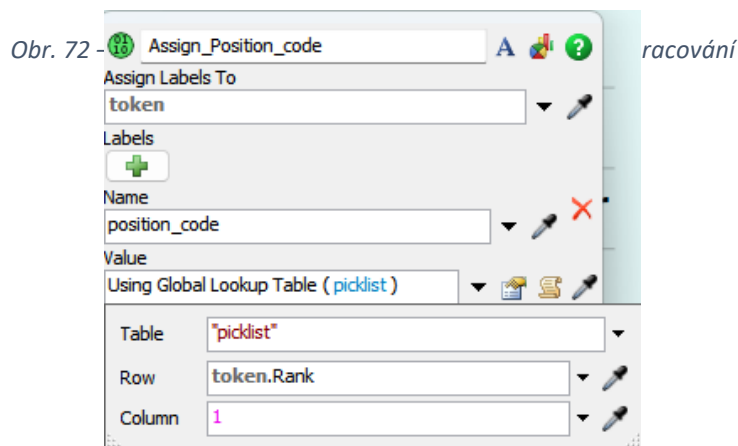


Při spuštění subprocesu dojde taktéž k vygenerování tokenů podle počtu dílů v daném picklistu.

Na začátku subprocesu přiřadí objekty Assign\_Position\_code, Assign\_Drawer\_size a Assign\_Item\_num (všechny typu Assign label) tokenu podle tabulky picklist labely s informacemi o dílu (pozice, velikost úložného boxu a číslo dílu), který se má vyskladnit



(viz 9.2.2). Labely jsou tokenům přiřazovány podle pořadí jejich vzniku (Rank), tzn. 1.

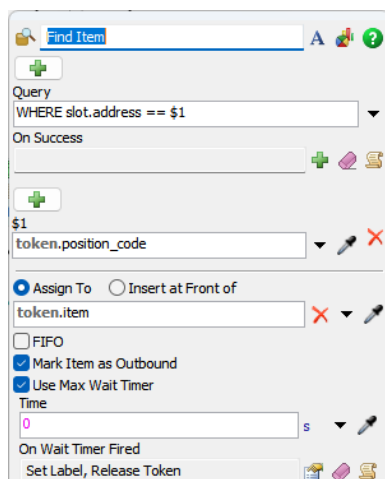


Obr. 73 - Přiřazení labelu „position\_code“ – pozice dílu, vlastní zpracování

token obsahuje informace z prvního řádku tabulky, 2. z druhého atd. Ukázka přiřazení labelu position\_code je zobrazena na Obr. 73.

Další krok je realizován objektem Acquire\_operator (Acquire Resource). Při vstupu tokenu do objektu je následně zavolán objekt Operator (Resource), který odkazuje na pracovníka (Operator) ve 3D modelu. Pracovník ve 3D modelu od této chvíle vykonává aktivity definované v Process flow.

Dále je objektem Find Item (Find Item) vyhledána pozice boxu, ve kterém se nachází vyskladňovaný díl. Pozice je vyhledána, pokud souhlasí label position\_code tokenu s adresou slotu (pozice umístění boxu) v některém z Racků (viz. Obr.74). V některých případech je díl dovážen z externího skladu, tudíž tuto pozici nelze nalézt v modelovém



Obr. 74 - Nastavení objektu Find Item, vlastní zpracování

skladu. Tyto pozice jsou uloženy do tabulky External\_WH (viz. Obr.75). Vyskladnění těchto pozic trvá vždy stejnou dobu definovanou objektem Delay\_external\_wh (Delay).

Item_position
OVEN_SUB
OVEN_SUB
OVEN_SUB
OVEN_SUB
OVEN_SUB
OVEN_SUB
OVEN_SUB
OVEN_SUB
OVEN_SUB

Obr. 75 - Tabulka External\_WH, vlastní zpracování

V případě že je pozice ve skladu nalezena, je tokenu přidělena objektem Assign\_Position\_height (Assign label) ještě informace o výšce skladovací pozice. Výška pozice je převzata ze souřadnic Flowitemu v 3D modelu a je uváděna v metrech. Tento údaj je později zapotřebí pro určení doby vyskladnění boxu. Zároveň je důležitou vstupní informací pro následující objekt Need\_a\_Ladder\_? (Decide), který rozhoduje, zda je pro vyskladnění pozice potřebný žebřík. Rozhodnutí je prováděno na základě porovnání Z-tové souřadnice Flowitemu a hodnoty parametru High\_position\_height (viz. Obr. 76). Pokud je Z-tová souřadnice Flowitemu větší než hodnota tohoto parametru, pak je tokenu přidělen label height = „High“, čímž je označeno že k vyskladnění je potřeba žebřík a přechází se na proces Picking w/ ladder (Pickování s použitím žebříku). Pokud je hodnota Z-tové souřadnice menší, přechází se na proces Picking w/o ladder (Pickování bez použití žebříku). Parametr High\_position\_height je primárně nastaven na hodnotu 2 metry, ale tuto hodnotu lze lehce změnit, např. když dojde k úpravě regálů.



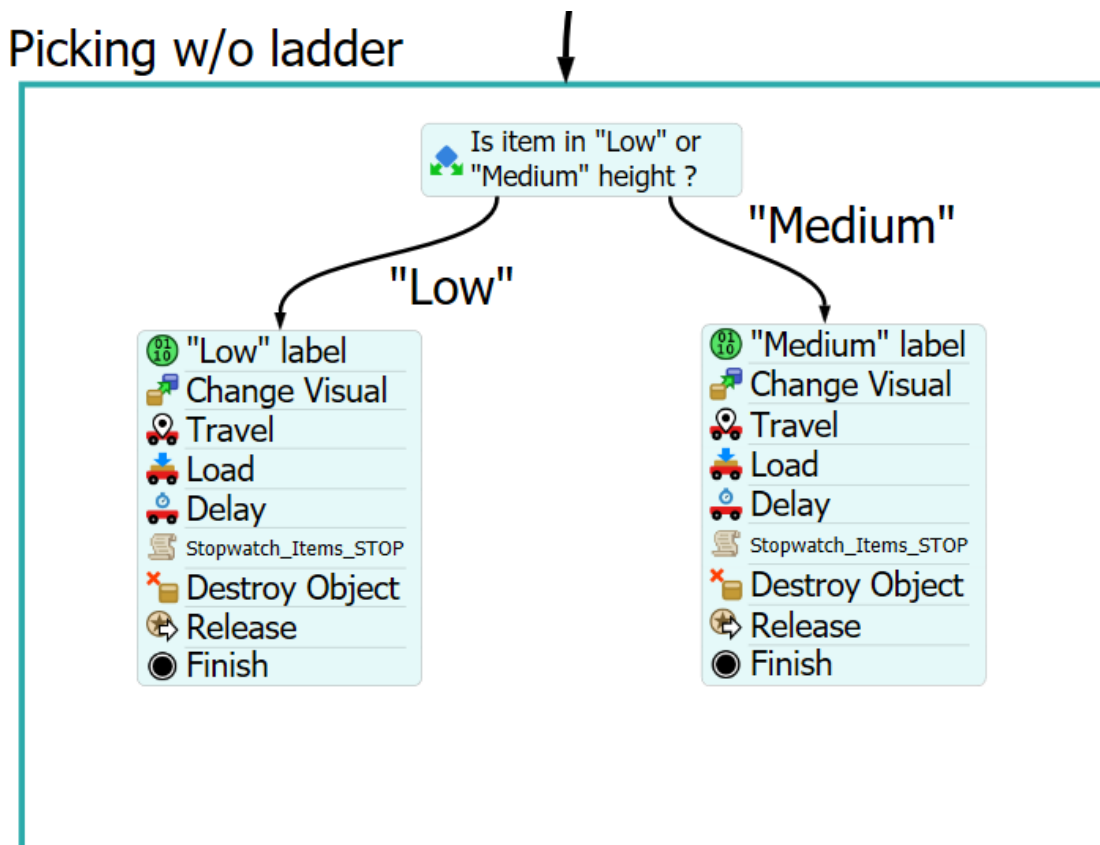
```

1 Object current = param(1);
2 treenode activity = param(2);
3 Token token = param(3);
4 treenode processFlow = ownerobject(activity);
5 /**popup:Conditional*/
6 /**Conditional Decide*/
7 /** \nIf some condition is true (equal to 1), send to a given connection/activity. Otherwise, send to a different connection/activity.*/
8
9 if (** \nCondition: **/**tag:expression/**/token.height <= Model.parameters.High_position_height/**/)
10     return /** \nIf Condition is true: **/**tag:true/**/1/**/;
11
12 return /** \nIf Condition is false: **/**tag:false/**/2/**/;
13

```

Obr. 76 - Nastavení objektu Need\_a\_Ladder\_?, vlastní zpracování

Proces Picking w/o ladder (Obr. 77) začíná rozhodovacím objektem typu Decide. Zde se obdobně porovnávají Z-tové souřadnice, tentokrát je ale rozhodováno, zda se musí pracovník pro box ohnout (pozice „Low“), či je ve výšce jeho pasu (pozice „Medium“) (viz. Obr. 78). Následující popis modelu je stejný pro vyskladnění boxů na pozici „Low“ i „Medium“. Rozdíl je pouze v labelu vyska přiděleném tokenu pomocí objektu Assign label, který je přidělen podle pozice boxu.



Obr. 78 – Schéma procesu „Picking w/o ladder“, vlastní zpracování

```

1 Object current = param(1);
2 treenode activity = param(2);
3 Token token = param(3);
4 treenode processFlow = ownerobject(activity);
5 /**popup:Conditional*/
6 /**Conditional Decide*/
7 /** \nIf some condition is true (equal to 1), send to a given connection/activity. Otherwise, send to a different connection/activity.*/
8
9 if (** \nCondition: ****tag:expression***/token.height <= Model.parameters.Low_position_height/**/)
10     return /** \nIf Condition is true: ****tag:true***/1/**/;
11
12 return /** \nIf Condition is false: ****tag:false***/2/**/;
13
  
```

Obr. 77 - Nastavení objektu typu Decide pro pozice „Low“ a „Medium“, vlastní zpracování

Po přidělení labelu, jsou příslušné tokeny i flowitemy (reprezentující boxy ve 3D modelu) pro větší přehlednost, barevně odlišeny („Low“ = zelená, „Medium“ = oranžová). Dále pokračují tokeny do objektu Travel (Travel To Location), který obstarává pohyb pracovníka (Operator) k pozici boxu. Poté co pracovník dojde na pozici boxu, se průchodem tokenu spouští aktivity v objektech Load (Load), Picking\_time (Delay) a Destroy Object (Destroy Object), které simulují vyskladnění dílu. Čas vyskladnění v objektu Picking\_time (Delay) je dán maticí z tabulky *Time\_matrix* (viz. Obr. 79) a je

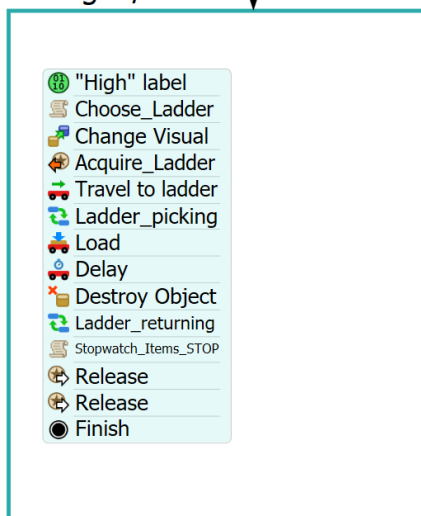
závislý na velikosti boxu, v němž se díl nachází a jeho poloze v regále. Po vyskladnění dílu přechází token do objektu Stopwatch\_Items\_STOP, který při jeho průchodu zapíše aktuální čas simulace do tabulky *Items\_cycle\_times* (do sloupce Time OUT) a zároveň vypočte a zapíše celkový čas vyskladnění daného dílu (sloupec Item\_cycle\_time). Poté je při vstupu tokenu do objektu Destroy Object (Destroy Object) “zničen” flowitemem nacházející se na příslušné pozici v Racku ve 3D modelu. Dále vstupuje token do objektu Release (Release Resource). Ten odebere tokenu referenci na pracovníka (Operator), což simuluje dokončení vyskladnění objednávky a uvolnění pracovníka v reálném systému.

	Low	Medium	High
Drawer (XS)	63.80	63.50	74
Drawer (M)	45.80	45.50	56
Drawer (S)	58.80	58.50	69
Blue box (S)	44.80	44.50	55
Blue box (L)	44.80	44.50	55
Drawer (L)	49.80	49.50	60
Cardbox (L)	58.80	58.50	69
Tube (L)	42.80	42.50	53
Tube (S)	42.80	42.50	53

Obr. 79 - Matice vyskladňovacích časů, vlastní zpracování

Jak již bylo řečeno, nachází-li se úložný box s potřebným dílem výš než 2 metry nad zemí, je k jeho vyskladnění potřeba použít žebřík. V Process flow je tento proces popsán blokem Picking w/ladder (viz. Obr. 80). Při vstupu tokenu do tohoto bloku je mu přidělen label vyska = „High“. Dále pokračuje token do objektu Choose\_Ladder (Custom Code) (viz. Obr. 81), který má na starosti výběr žebříku použitého k vyskladnění. Žebříky jsou umístěny na 4 pozicích (viz. Kap. 8.1) a pracovník si vždy vybere ten, který je k němu v daný moment nejbližší.

Picking w/ ladder ↷



Obr. 80 - Blok Picking w/ ladder, vlastní zpracování

```

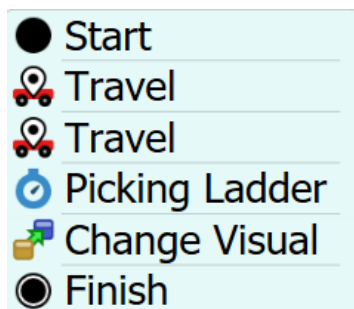
Choose_Ladder - Custom Code*
1 /**Custom Code*/
2 Object current = param(1);
3 treenode activity = param(2);
4 Token token = param(3);
5 treenode processFlow = ownerobject(activity);
6
7 Object chosen;
8 double min = 1000000;
9 Group ladders = Group("Ladder_group");
10 Array laddersArr = ladders.toFlatArray();
11 Object operator = token.resource;
12 for (int i = 1; i <= laddersArr.length; i++)
13 {
14     double distancetravel = distancetotravel(operator,laddersArr[i]);
15     print("distance of ", laddersArr[i], " is: ", distancetravel);
16     if (distancetravel < min)
17     {
18         chosen = laddersArr[i];
19         min = distancetravel;
20     }
21 }
22 token.ladder_closest = chosen;

```

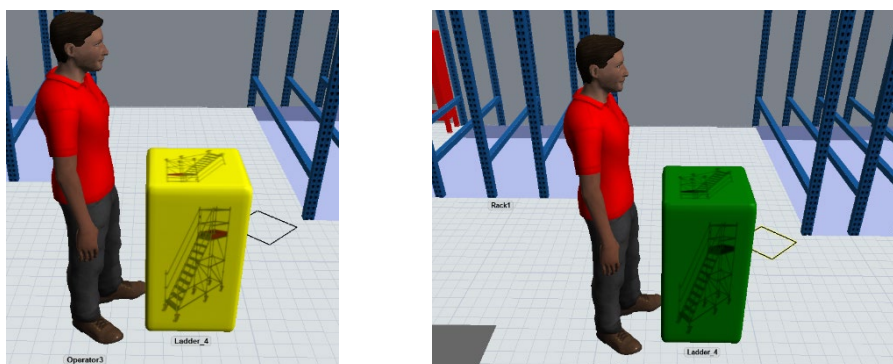
Obr. 81 – Kód objektu „Choose\_Ladder“ pro výběr žebříku, vlastní zpracování

Po výběru žebříku dojde stejně jako u pozic „Low“ a „Medium“ ke změně barvy flowitemů (boxů) a tokenů, v tomto případě na červenou. Dalším krokem je přidělení žebříku k příslušnému tokenu objektem Acquire\_Ladder (Acquire Resource). Tím je zajištěné, že pro vyjmutí boxu z dané pozice bude použit příslušný žebřík. Objekt Travel\_to\_ladder poté zajistí přesun pracovníka k žebříku. Následující subproces

Ladder\_picking (Sub Flow) (Obr.82) reprezentuje proces vzetí a transportu žebříku k pozici vyskladňovaného boxu. Doba vzetí žebříku je stanovena na 15 vteřin a doba transportu je automaticky dopočítána softwarem na základě vstupních parametrů. Pro lepší orientaci je aktuálně používaný žebřík v 3D modelu zbarven do zelena (viz. Obr. 83).



Obr. 83 - Objekty subprocesu Ladder\_picking, vlastní zpracování



Obr. 82 - Barevné rozlišení žebříků – vlevo: neaktivní žebřík, vpravo: aktivní žebřík, vlastní zpracování

Poté přechází pracovník i se žebříkem k pozici dílu, jehož vyskladnění je simulováno objektem Delay (Delay). Poté je při vstupu tokenu do objektu Destroy Object (Destroy Object) “zničen” flowitem nacházející se na příslušné pozici v Racku ve 3D modelu. Po vyskladnění dílu je žebřík pracovníkem opět vrácen na původní pozici. Poté již token projde skrze objekt Stopwatch\_Items\_STOP, který vypočítá čas vyskladnění daného dílu a zapíše jej do tabulky *Items\_cycle\_times* (Obr.84) (viz. Odstavec Proces Picking w/o ladder) a 2 objekty typu Release Resource, které odeberou tokenu referenci na pracovníka a použitý žebřík, aby mohli být využiti dále.

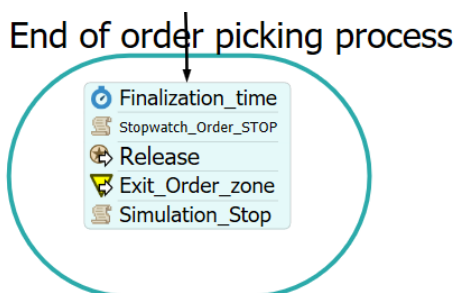
Order	Item_num	Time IN [s]	Time OUT [s]	Item_cycle_time [min]
MLE284401	800-1942-01	7260.16	7325.49	1.09
MLE226590	82206	7260.18	7320.80	1.01
MLE275753	800-2590-01	7260.21	7312.84	0.88
MLE291583	800-3293-01	7261.04	7326.80	1.10
MLE291582	800-3293-01	7261.43	7376.20	1.91
MLE285397	800-2567-01	7261.98	7330.53	1.14
MLE291584	800-3293-01	7262.22	7489.05	3.78
MLE281408	38011-1	7262.52	7319.91	0.96

Obr. 84 - Tabulka Items\_cycle\_times, vlastní zpracování

Token se po vykonání všech akcí v objektech Picking w/o ladder a Picking w/ladder přesouvá dále do konečného bloku End of order picking process, kde jsou prováděny konečné aktivity.

### 9.3.4 End of order picking process

V tomto bloku (Obr. 85) dochází k finalizaci procesu vyskladnění objednávky. Při vstupu tokenu do bloku se nejprve spustí objekt Finalization\_time (Delay) reprezentující dokončovací administrativní práce (kontrola picklistu, podpis ...).



Obr. 85 - Blok End of order picking process, vlastní zpracování

Poté se token přesouvá do objektu Stopwatch\_Order\_STOP (Custom Code) (Obr. 86). Ten zajišťuje že se aktuální čas simulace zapíše do tabulky Orders\_cycle\_time, jako výstupní čas objednávky (Time OUT) (viz. Kap. 9.3.2, Obr. 68). Zároveň zde dochází k výpočtu celkového času vyskladnění objednávky (Cycle\_time) a jeho zápisu do téže tabulky. Hodnoty jsou zapisovány na základě labelu time\_table\_ref, tudíž jsou vždy



zapsány do stejného řádku jako počáteční čas v bloku Stopwatch\_Order\_START (viz. Kap. 9.3.2).

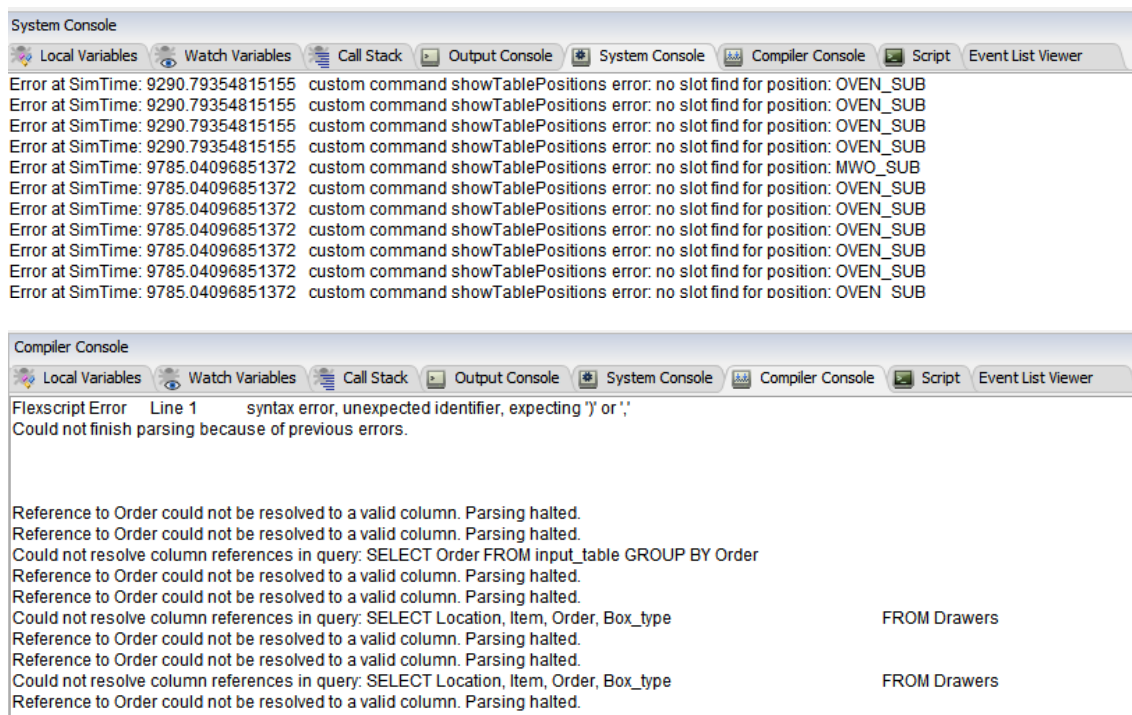
```
Stopwatch_Order_STOP - Custom Code
1 | **Custom Code*/
2 | Object current = param(1);
3 | treenode activity = param(2);
4 | Token token = param(3);
5 | treenode processFlow = ownerobject(activity);
6 |
7 | Table table = Table("Orders_cycle_times");
8 | table[token.time_table_ref][3] = Model.time; //tady uloz nalezve te pozice
9 |
10 | if (((table[token.time_table_ref][3] - table[token.time_table_ref][2])/60) > 960)
11 | {
12 |     table[token.time_table_ref][4] = ((table[token.time_table_ref][3] - table[token.time_table_ref][2])/60) - 960;
13 | }
14 | else
15 | {
16 |     table[token.time_table_ref][4] = ((table[token.time_table_ref][3] - table[token.time_table_ref][2])/60);
17 | }
18 |
```

Obr. 86 – Programový kód „Stopwatch\_Order\_STOP“, vlastní zpracování

Dále je při průchodu blokem Release (Release Resource) odebrána hodnota labelu odkazujícího na pracovníka zpracovávajícího danou objednávku. To reprezentuje situaci, kdy reálný pracovník dokončil danou objednávku a může začít pracovat na nové. Dále již token projde přes objekt Exit\_Order\_zone (Exit Zone) až do objektu Simulation\_Stop (Custom Code), který po průchodu všech tokenů (objednávek) zastaví simulaci.

### 9.3.5 Verifikace

Verifikace modelu probíhala vizuální kontrolou výsledného modelu a výsledků při pokusné simulaci částečně ve spolupráci se zadavatelem. K debugování modelu byl využit vestavěné nástroje System Console a Compiler Console (viz. Obr.87). V rámci verifikace nebyly nalezeny žádné nesrovnalosti.



The image shows two screenshots of software consoles. The top screenshot is the 'System Console' showing a series of error messages: 'Error at SimTime: 9290.79354815155 custom command showTablePositions error: no slot find for position: OVEN\_SUB'. The bottom screenshot is the 'Compiler Console' showing a 'Flexscript Error' on line 1: 'syntax error, unexpected identifier, expecting ')' or ';''. Below this, it states 'Could not finish parsing because of previous errors.' and lists several 'Reference to Order could not be resolved to a valid column. Parsing halted.' messages, along with a query: 'SELECT Location, Item, Order, Box\_type FROM Drawers'.

Obr. 87 - Ukázka chybových hlášek v System Console (nahore) a Compiler Console (dole), vlastní zpracování

### 9.3.6 Validace

Modelový sklad je oproti reálnému systému zjednodušen v několika ohledech. Pokud dochází v reálném systému k vyskladnění více dílů z vyšších pozic, pracovníci žebřík nevrací zpět výchozí pozici po vyskladnění každého dílu, ale používají ho, dokud nevyskladní všechny díly a až poté ho odvezou na výchozí pozici. V modelu byl tento postup z časových důvodů zjednodušen a pracovníci vždy po vyskladnění dílu vracejí žebřík zpět na místo.

Dále bylo zjednodušeno vyskladnění dílů, které se reálném systému dovážejí z externího skladu, nacházejícího se v jiné části závodu. Tento sklad není zakomponován ve 3D modelu. Vyskladnění dílů je řešeno pouze v prostředí Process Flow, kde byla doba vyskladnění dílu stanovena podle dat z podnikového systému na 2 minuty. Tyto skutečnosti ovšem nijak vážně neovlivňují chod simulace ani její výsledky.

Výsledky simulace byly konzultovány a kontrolovány ve spolupráci se zaměstnanci společnosti a odpovídají reálnému stavu. Simulační model tedy odpovídá reálnému systému.

## 10 Experimentální část

V této části byly provedeny simulační běhy pro 3 scénáře – současný stav a 2 nově navržené varianty uspořádání položek ve skladu. Při simulacích byly sledovány ukazatele definované v kap.7. Výsledky jednotlivých variant jsou pro lepší přehlednost zpracovány do tabulek.

### 10.1 Vstupní data

Pro potřeby simulace v této práci byl použit soubor reálných dat vygenerovaných podnikovým systémem. Celkem se jedná o set 575 objednávek složených z 6975 dílů. V souboru se nacházejí údaje o objednávkách, vyskladňovaných dílech (čas vyskladnění a počet vyskladněných dílů) a jejich umístění (ukázka obsahu souboru viz. Obr 88). Data jsou seřazena vzestupně podle času vyskladnění dílů. Pro varianty 1 a 2 byly vytvořeny modifikace základního souboru, kde byly přemístěny pozice některých dílů. Přesuny jsou popsány podrobněji u každé z variant.

Order	Due date	Item	Location	Sum of Qty
ABC123456	16.02.2024	123-450	UC0302C02	1
ABC123456	16.02.2024	54696	UD0108C01	1
ABC123456	16.02.2024	645-844	UD0108C01	1
ACG658651	19.02.2024	944-385	UC1202C01	5
ACG658651	19.02.2024	899-1458	J04003	2
ACG658651	19.02.2024	949-488	OVEN_SUB	1
RDA644895	23.02.2024	186-168	UA0112C02	1

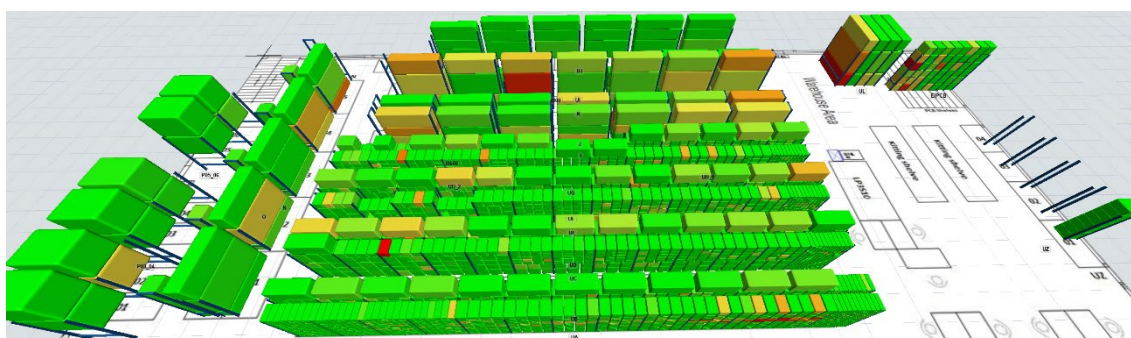
Obr. 88 – Ukázka uspořádání tabulky vstupních dat, vlastní zpracování

## 10.2 Současný stav

V této variantě byla použita neupravená data vygenerovaná z podnikového systému dle skutečnosti. V rámci simulace byl sledován celkový čas dokončení setu objednávek, průměrný čas vyskladnění dílu z pozice, počet dílů vzatých z neergonomických pozic a další viz. Tab. 4. Výsledky simulace jsou viditelné v Tab.4. Pro analýzu a vizualizace počtu dílů ležících v neergonomických pozicích byl využit simulační model č. 1. Výsledky z tohoto modelu jsou zobrazeny na Obr. 89 Barevné rozlišení bylo provedeno dle barevné škály zobrazené na Obr. 54 v Kap 9.1.

*Tab. 4 - Výsledky simulace – současný stav, vlastní zpracování*

Metrika	Průměrná hodnota	Jednotky
<b>Celkový čas dokončení zkušebního setu objednávek</b>	76,42	h/set
<b>Průměrný čas vyskladnění dílu z pozice</b>	60,24	s/pozice
<b>Počet dílů vzatých z neergonomických pozic</b>	120	ks
<b>Průměrné využití pracovníků</b>	74	%
<b>Počet zpracovaných objednávek za 1 týden</b>	819	
<b>Průměrná uražená vzdálenost</b>	4569,51	m/pracovník



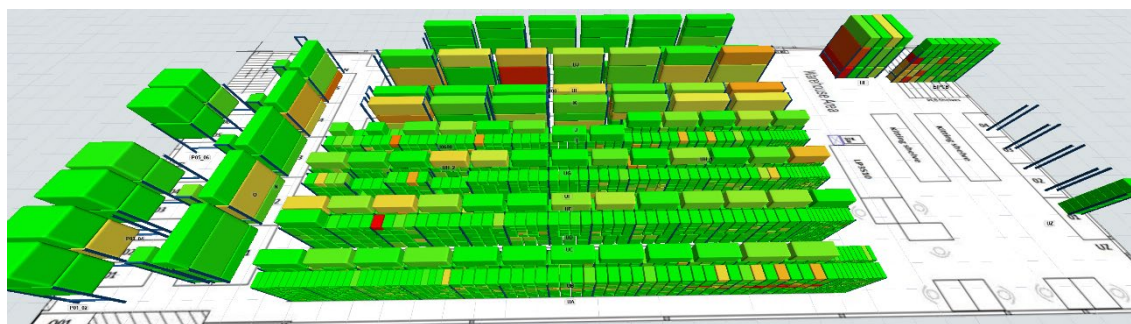
*Obr. 89 - Zobrazení počtu picků – současný stav, vlastní zpracování*

## 10.1 Varianta 1

Vstupem pro simulaci varianty 1 byla upravená vstupní data původního stavu. Nejprve bylo určeno prvních 20 % pozic, ze kterých jsou díly nejčastěji vyskladňovány. Dále bylo nutné analyzovat, které pozice z nich jsou již v neergonomické výšce (2 a více m nad zemí). Díly z těchto pozic byly následně prohozeny s díly na nižších pozicích, které nejsou tak často vyskladňovány. Podmínkou společnosti bylo že boxy, ve kterých jsou díly uloženy jsou stejné velikosti a nacházejí se ve stejném regále. Výsledky s upravenými daty jsou zobrazeny v Tab. 5. Vizuální zobrazení za využití modelu č.1 je poté zobrazeno na Obr. 90.

Tab. 5 - Výsledky simulace – varianta 1, vlastní zpracování

Metrika	Průměrná hodnota	Jednotky
Celkový čas dokončení setu objednávek	76,36	h/set
Průměrný čas vyskladnění dílu z pozice	59,82	s/pozice
Počet dílů vzatých z neergonomických pozic	86	ks
Průměrné využití pracovníků	74	%
Počet zpracovaných objednávek za 1 týden	831	
Průměrná uražená vzdálenost	4497,63	m/pracovník



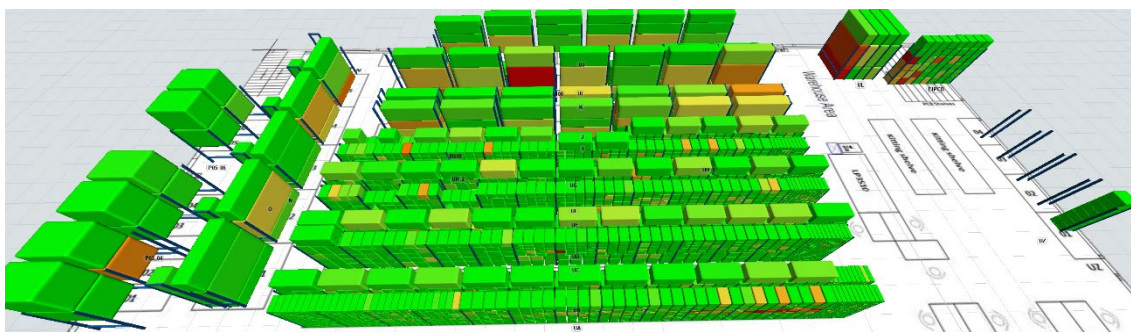
Obr. 90 - Zobrazení počtu picků – varianta 1, vlastní zpracování

## 10.2 Varianta 2

Vstupem pro simulaci varianty 2 byla upravená vstupní data původního stavu. Oproti variantě 1 zde bylo analyzováno prvních 40 % pozic, ze kterých jsou díly nejčastěji vyskladňovány. Dále bylo nutné z těchto pozic vybrat ty pozice, které jsou stále v neergonomické výšce (2 a více m nad zemí). Díly z těchto pozic byly následně prohozeny s díly na nižších pozicích. Podmínkou v této variantě bylo pouze že boxy, ve kterých jsou díly uloženy jsou stejné velikosti. Výsledky s upravenými daty jsou zobrazeny v Tab. 6. Vizualní zobrazení za využití modelu č.1 je poté zobrazeno na Obr. 91.

*Tab. 6 - Výsledky simulace – varianta 2, vlastní zpracování*

Metrika	Průměrná hodnota	Jednotky
<b>Celkový čas dokončení setu objednávek</b>	76,19	h/set
<b>Průměrný čas vyskladnění dílu z pozice</b>	59,60	s/pozice
<b>Počet dílů vzatých z neergonomických pozic</b>	26	ks
<b>Průměrné využití pracovníků</b>	74	%
<b>Počet zpracovaných objednávek za 1 týden</b>	837	
<b>Průměrná uražená vzdálenost</b>	4152,54	m/pracovník



*Obr. 91 - Zobrazení počtu picků - varianta 2, vlastní zpracování*

## 11 Vyhodnocení výsledků

V této kapitole jsou vyhodnoceny a porovnány výsledky simulačních běhů jednotlivých variant uspořádání dílů ve skladu. Obě nové varianty uspořádání jsou porovnány vůči současnému stavu i vůči sobě.

### 11.1 Porovnání současného stavu a varianty 1

V této podkapitole je porovnán současný stav s 1. variantou nového uspořádání dílů ve skladu. Ve variantě 1 bylo přesunuto 10 dílů na nové ergonomičtější pozice. Porovnání výsledků obou simulačních studií je zobrazeno v Tab. 7. Procentuální rozdíl byl vypočten jako procentuální změna varianty 1 vůči současnému stavu (současný stav = 100 %).

Tab. 7 - Porovnání výsledků simulace pro současný stav a variantu 1, vlastní zpracování

Metrika	Současný stav	Varianta 1	% Rozdíl	Abs. Rozdíl	Jednotky
Celkový čas dokončení setu objednávek	76,42	76,36	-0,1%	0,06	h/set
Průměrný čas vyskladnění dílu z pozice	60,24	59,82	-0,7%	0,42	s/pozice
Počet dílů vzatých z neergonomických pozic	120	86	-28,33%	34,00	ks
Průměrné využití pracovníků	74	74	0,0%	0,00	%
Počet zpracovaných objednávek za 1 týden	819	831	1,5%	12,00	-
Průměrná uražená vzdálenost	4569,51	4497,63	-1,6%	71,88	m/pracovník

Nejvýraznější změna nastala v počtu dílů vyskladňovaných z neergonomických pozic – pokles o 28,33 % = 34ks na 86 dílů. Zároveň s tím došlo k poklesu průměrného času vyskladnění dílu (o 0,7 % = 0,42s na 59,82 sekundy) a celkového času dokončení setu objednávek (o 0,1 % = 0,06h na 76,36 h). V důsledku snížení těchto časů došlo k navýšení počtu objednávek zpracovaných za jeden týden o 1,5 % = 12 na 831. Vytížení operátorů zůstalo beze změny 74 %. Zkrátila se však vzdálenost nachozená operátory o 1,6 % = 71,88 m na 4497,63 m.

## 11.2 Porovnání současného stavu a varianty 2

Oproti současnému stavu bylo ve variantě 2 přesunuto 20 dílů na nové, více ergonomické pozice. Porovnání obou variant je zobrazeno v Tab. 8. Procentuální rozdíl byl vypočten jako procentuální změna varianty 2 vůči současnému stavu (současný stav = 100 %).

Tab. 8 - Porovnání výsledků simulace pro současný stav a variantu 2, vlastní zpracování

Metrika	Současný stav	Varianta 2	% Rozdíl	Abs. Rozdíl	Jednotky
Celkový čas dokončení setu objednávek	76,42	76,19	-0,3%	0,23	h/set
Průměrný čas vyskladnění dílu z pozice	60,24	59,6	-1,1%	0,64	s/pozice
Počet dílů vzatých z neergonomických pozic	120	26	-78,33%	94,00	ks
Průměrné využití pracovníků	74	74	0,0%	0,00	%
Počet zpracovaných objednávek za 1 týden	819	837	2,2%	18,00	-
Průměrná uražená vzdálenost	4569,51	4152,54	-9,1%	416,97	m/pracovník

Z výsledků je patrné že se počet dílů vyskladňovaných z neergonomických pozic snížil o 78,33 % = 94ks, což je o více než o 3/4. Časový rozdíl dokončení setu objednávek byl v porovnání o 0,3% = 0,23h nižší a stejně tak průměrný čas vyskladnění dílu klesl o 1,1 % = 0,64 s. Počet vyskladněných objednávek za 1 týden se zvýšil o 2,2 % = 18 na 837 objednávek. Využití pracovníků zůstalo opět beze změny, ale jejich průměrně uražená vzdálenost se zkrátila o 9,1 % = 416,97 m na 4152,54 m.

## 11.3 Porovnání varianty 1 a varianty 2

V Tab. 9 je zobrazeno porovnání výsledků obou variant. Procentuální rozdíl byl vypočten jako procentuální změna varianty 2 vůči variantě 1 (varianta 1 = 100 %).

Tab. 9 - Porovnání výsledků simulace pro variantu 1 a variantu 2, vlastní zpracování

Metrika	Varianta 1	Varianta 2	% Rozdíl	Abs. Rozdíl	Jednotky
Celkový čas dokončení setu objednávek	76,36	76,19	-0,2%	0,17	h/set
Průměrný čas vyskladnění dílu z pozice	59,82	59,6	-0,4%	0,22	s/pozice
Počet dílů vzatých z neergonomických pozic	86	26	-69,77%	60,00	ks
Průměrné využití pracovníků	74	74	0,0%	0,00	%
Počet zpracovaných objednávek za 1 týden	831	837	0,7%	6,00	-
Průměrná uražená vzdálenost	4497,63	4152,54	-7,7%	345,09	m/pracovník

Nejvýraznější změna nastala opět v počtu dílů vyskladňovaných z neergonomických pozic – pokles o 69,77 % na 26 dílů. Zároveň s tím došlo k poklesu průměrného času vyskladnění dílu (o 0,4 % = 0,22s na 59,6 sekundy) a celkového času dokončení setu objednávek (o 0,2 % = 0,17 h na 76,19 h). V důsledku snížení těchto časů došlo k navýšení





počtu objednávek zpracovaných za jeden týden o 0,7 % na 837. Vytížení operátorů zůstalo opět beze změny 74 %. Zkrátila se však vzdálenost nachozená operátory o 7,7 % = 345,09 m na 4330,24 m.

## 11.4 Závěrečné vyhodnocení

Z výsledků simulací je patrné že nejlepších výsledků bylo dosaženo ve variantě 2. Počet neergonomických pozic se při tomto uspořádání snížil o 78,33 % oproti současnému stavu a o 69,77 % oproti variantě 1. Časová úspora oproti původnímu stavu je 0,23 h na zkušebním setu objednávek. Časová úspora na vyskladnění 1 dílu je poté v průměru 0,64s. Při novém uspořádání jsou pracovníci schopni vyskladnit o 18objednávek za týden více než při současném stavu. Vytížení pracovníků zůstává stejné, avšak jejich uražená vzdálenost se z důvodu nižšího používání žebříků snížila v průměru o 9,1 %.

## Závěr

Cílem této práce bylo vytvoření simulačního modelu a návrh optimalizace uspořádání položek pro výrobní sklad společnosti Safran Cabin. Na základě požadavků, diskusí se zaměstnanci společnosti a dat z reálného provozu byl vytvořen koncepční model skladu. Z koncepčního modelu byly následně pomocí simulačního softwaru FlexSim vytvořeny 2 simulační modely. První model slouží pro rychlou analýzu počtu vyskladnění jednotlivých dílů ze skladu. Druhý model je poté postaven jako nástroj pro použití při optimalizaci skladu, pomocí něhož lze snadno simulovat nové návrhy uspořádání položek a vyhodnocením definovaných ukazatelů výkonnosti je vzájemně porovnávat.

Ukazatele výkonnosti byly definovány na základě domluvy se společností tak, aby pro ně měly, pokud možno co největší význam. Celkem tak bylo definováno 6 ukazatelů výkonnosti: celkový čas dokončení zkušebního setu objednávek, průměrný čas vyskladnění dílu z pozice, počet odebraných dílů z neergonomických pozic, průměrné využití pracovníků, průměrná vzdálenost uražená pracovníky při přípravě setu objednávek a počet objednávek zpracovaných za týden. Tyto indikátory slouží jako výstup simulačních běhů.

V rámci optimalizace byly navrženy 2 nové varianty uspořádání položek. Obě varianty byly porovnávány se současným stavem a v obou byly přemísťovány pouze díly z nejvyšších pozic, pro které je zapotřebí využít žebřík. Ve variantě 1 bylo přesunuto 10 boxů s díly na nové pozice. Tato úprava zajistila zkrácení času dokončení setu objednávek o 0,06 h/set, snížení průměrného času vyskladnění dílu z pozice o 0,42 s/pozici, snížení počtu dílů v neergonomických pozicích o 34 ks a zkrácení průměrné vzdálenosti uražené pracovníkem o 71,88 m. Počet objednávek zpracovaných za jeden týden se zvýšil o 12 oproti současnému stavu. Ve variantě 2 bylo přesunuto 20 boxů s díly do nových pozic. Tato úprava zajistila zkrácení času dokončení setu objednávek o 0,23 h/set, snížení průměrného času vyskladnění dílu z pozice o 0,64 s/pozici, snížení počtu dílů v neergonomických pozicích o 94 ks a zkrácení průměrné vzdálenosti uražené



pracovníkem o 416,97 m. Počet objednávek zpracovaných za jeden týden se zvýšil o 18 oproti současnému stavu.

Z výsledků může vyplývat, že nedošlo k příliš velkému zlepšení procesu, hlavně v časech vyskladnění dílů. Pokud ovšem vztáhneme toto zlepšení na větší množství vyskladněných dílů, např. za 1 rok je to již nezanedbatelná hodnota. Výsledně tedy doporučuji variantu 2, která se na základě výsledků všech ukazatelů jeví jako nejefektivnější. Řešený skladový proces by bylo možné dále optimalizovat např. lepším uspořádáním pracoviště, či vyřešením technických problémů (např. zlepšení internetového připojení pro čtečky čárových kódů), které celý proces značně zpomalují.

## Seznam obrázků

Obr. 1 - Klíčové komponenty distribuce a logistiky.....	15
Obr. 2 - Logistický řetězec .....	16
Obr. 3 - Spaghetti diagram .....	19
Obr. 4 - - Sankeyův diagram .....	20
Obr. 5 - Postupový (vývojový) diagram .....	21
Obr. 6 - VSM mapa .....	21
Obr. 7 - Schéma systému hlubinného zakládání.....	27
Obr. 8 - Schéma kompaktního úložného systému .....	27
Obr. 9 - Schéma karuselového systému.....	28
Obr. 10 - Schéma skladu s autonomními zakladači.....	29
Obr. 11 – Sam-Loydovo puzzle.....	30
Obr. 12 - Schéma skladového systému na základě Sam Loydova puzzle.....	30
Obr. 13 - Schéma robotického skladovacího systému .....	31
Obr. 14 - Ukázka mobilního robotického skladového systému .....	32
Obr. 15 - Nejčastější typy rozvržení skladů .....	36
Obr. 16 - Nejčastěji používané typy vyskladňovacích cest.....	38
Obr. 17 - Obecný průběh simulace .....	40
Obr. 18 – Ukázka prostředí softwaru Arena .....	45
Obr. 19 - Ukázka prostředí Plant Simulation.....	46
Obr. 20 - Ukázka prostředí FlexSim .....	46
Obr. 21 - Ukázka prostředí Visual Components.....	47



Obr. 22 - Schéma tvorby simulačního projektu .....	51
Obr. 23 - Proces tvorby simulačního modelu.....	54
Obr. 24 - Interiér pro Business jet.....	57
Obr. 25 - Kombinovaná trouba pro Business jet.....	58
Obr. 26 - Záložka "Model" .....	59
Obr. 27 - Prostředí Process Flow .....	60
Obr. 28 - Ukázka objektů Process Flow .....	61
Obr. 29 - Objekt „Source“ .....	62
Obr. 30 - Objekt „Queue“ .....	63
Obr. 31 - Objekt „Task Executer“ .....	63
Obr. 32 - Objekt „Operator“ .....	64
Obr. 33 - Objekt „Rack“ .....	64
Obr. 34 - Typy objektu „Source“ .....	65
Obr. 35 - Objekt „Assign Labels“ .....	65
Obr. 36 - Objekt „Create Tokens“ .....	66
Obr. 37 - Objekt „Custom Code“ .....	66
Obr. 38 - Objekt „Decide“ .....	66
Obr. 39 - Objekty „Zone“ .....	67
Obr. 40 - Objekty „Resource“ .....	67
Obr. 41 - Objekty „Objects“ .....	68
Obr. 42 - Objekt „Delay“ .....	68
Obr. 43 - Objekty „Task Sequences“ .....	68



Obr. 44 - Objekty „Warehousing“ .....	69
Obr. 45 - Objekty „Sub-flow“ .....	69
Obr. 46 - - Schéma rozmístění regálů ve skladu.....	72
Obr. 47 - Schéma umístění žebříku ve skladu .....	73
Obr. 48 - Ukázka seřazení dat – simulace počtu picků.....	74
Obr. 49 - Ukázka seřazení dat – simulace procesu vyskladňování.....	75
Obr. 50 - Ukázka dat – aktivity v procesu vyskladňování.....	75
Obr. 51 - Model č.1 - 3D model.....	76
Obr. 52 Programový kód objektu Warehouse_generate_Col_by_Picks.....	77
Obr. 53 – Blok „Items_generation“ .....	77
Obr. 54 - Barevná škála pro počet picků .....	78
Obr. 55 - - Ukázka rozmístění skladových pozic (slotů) regálu EIPCB .....	79
Obr. 56 - Jeden ze 4 typů mapovacích schémat použitých v modelu.....	79
Obr. 57 - Časový rozvrh pracovníků: tmavě fialová – čas mimo pracovní dobu, světle modrá – ranní meetingy, světle fialová – obědová pauza, oranžová, pracovní pauza ..	80
Obr. 58 - Rychlost pohybu se žebříkem .....	81
Obr. 59 - Ukázka bariér objektu A* Navigator: šedá – schodiště, světle modrá – prostor regálů .....	81
Obr. 60 - Přehled jednotlivých bloků Process Flow pro 2. model.....	83
Obr. 61 - Blok Orders_creation .....	84
Obr. 62 – Programový kód objektu Table_distinct .....	84
Obr. 63 - Nastavení objektu Creating_orders .....	85
Obr. 64 – Tabulka orders_list .....	85



Obr. 65 – Vstupní tabulka input_table.....	85
Obr. 66 - Blok Orders prepair and picking.....	86
Obr. 67 – Programový kód objektu Stopwatch_Order_START .....	87
Obr. 68 - Tabulka Orders_cycle_times .....	87
Obr. 69 - Programový kód objektu Generate_objects.....	88
Obr. 70 - Tabulka picklist.....	88
Obr. 71 - Programový kód objektu Generate_picklist .....	88
Obr. 72 – Schéma subprocessu „Order Sub Flow“ .....	89
Obr. 73 - Přiřazení labelu „position_code“ – pozice dílu .....	89
Obr. 74 - Tabulka External_WH.....	90
Obr. 75 - Nastavení objektu Find Item.....	90
Obr. 76 - Nastavení objektu Need_a_Ladder_?.....	91
Obr. 77 - Nastavení objektu typu Decide pro pozice „Low“ a „Medium“ .....	92
Obr. 78 – Schéma procesu „Picking w/o ladder“ .....	92
Obr. 79 - Matice vyskladňovacích časů .....	93
Obr. 80 – Kód objektu „Choose_Ladder“ pro výběr žebříku.....	94
Obr. 81 - Blok Picking w/ ladder.....	94
Obr. 82 - Barevné rozlišení žebříků – vlevo: neaktivní žebřík, vpravo: aktivní žebřík .....	95
Obr. 83 - Objekty subprocessu Ladder_picking.....	95
Obr. 84 - Tabulka Items_cycle_times.....	96
Obr. 85 - Blok End of order picking process.....	96
Obr. 86 – Programový kód „Stopwatch_Order_STOP“ .....	97



---

Obr. 87 - Ukázka chybových hlášek v System Console (nahore) a Compiler Console (dole).....	98
Obr. 88 – Ukázka uspořádání tabulky vstupních dat .....	99
Obr. 89 - Zobrazení počtu picků – současný stav.....	100
Obr. 90 - Zobrazení počtu picků – varianta 1.....	101
Obr. 91 - Zobrazení počtu picků - varianta 2.....	102





## Seznam tabulek

Tab. 1 – Přehled ukazatelů výkonnosti ve skladech.....	35
Tab. 2 - Přehled metod využívaných při návrhu výrobních systémů .....	49
Tab. 3 - Metriky simulačního projektu .....	70
Tab. 4 - Výsledky simulace – současný stav .....	100
Tab. 5 - Výsledky simulace – varianta 1 .....	101
Tab. 6 - Výsledky simulace – varianta 2 .....	102
Tab. 7 - Porovnání výsledků simulace pro současný stav a variantu 1 .....	103
Tab. 8 - Porovnání výsledků simulace pro současný stav a variantu 2 .....	104
Tab. 9 - Porovnání výsledků simulace pro variantu 1 a variantu 2 .....	104

## Seznam použité literatury

- [1] RUSHTON, Alan; CROUCHER, Phil a BAKER, Peter. *The handbook of logistics and distribution management*. 5th ed. London: Chartered Institute of Logistics and Transport, 2014. ISBN 9780749466275.
- [2] BAKEŠOVÁ, Miroslava a KŘEŠŤAN, Vladimír. *Základy logistiky*. Jihlava: Vysoká škola polytechnická Jihlava, 2008. ISBN 978-80-87035-08-5.
- [3] BOBÁK, Roman. *Základy logistiky*. Brno: Vysoké učení technické, 1999. ISBN 80-214-1428-6.
- [4] JUROVÁ, Marie. *Výrobní a logistické procesy v podnikání*. Expert (Grada). Praha: Grada Publishing, 2016. ISBN 978-80-247-5717-9.
- [5] GROS, Ivan. *Velká kniha logistiky*. Praha: Vysoká škola chemicko-technologická v Praze, 2016. ISBN 978-80-7080-952-5.
- [6] SCULLIN, Chris. *The 6 Types of Warehouse Storage Systems*. Online. 2024. Dostupné z: <https://www.camcode.com/blog/types-of-warehouse-storage-systems/>. [cit. 2024-04-27].
- [7] JUNGHEINRICH. *Regály a regálové systémy*. Online. 2024. Dostupné z: <https://www.jungheinrich.cz/produkty/regaly>. [cit. 2024-04-27].
- [8] AZADEH, Kaveh; DE KOSTER, René; ROY, Debjit. Robotized and automated warehouse systems: Review and recent developments. *Transportation Science*, 2019, 53.4: 917-945.
- [9] KARÁSEK, Jan. An overview of warehouse optimization. *International journal of advances in telecommunications, electrotechnics, signals and systems*, 2013, 2.3: 111-117.
- [10] FRAZELLE, Edward H. *World-class warehousing and material handling*. 2nd ed. McGraw-Hill Education, 2016. ISBN 978-0-07-184283-9.
- [11] CRBGroup. Online. Dostupné z: <https://www.crbgroup.com/insights/consulting/warehouse-optimization>. [cit. 2024-03-26].
- [12] AnyLogic. Online. Dostupné z: <https://www.anylogic.com/blog/using-simulation-to-create-the-optimal-warehouse-layout/>. [cit. 2024-03-26].
- [13] BANKS, Jerry. *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*. USA: Engineering & Management Press, 1998. ISBN 0-471-13403-1
- [13] MANLIG, František. *Využití počítačové simulace výrobních systémů*. Liberec: TU v Liberci, 2014. ISBN 978-80-7494-162-7
- [14] LAW, Averill M. *Simulation Modeling and Analysis*. 5th ed. McGraw-Hill Education, 2015. ISBN 978-0-07-340132-4.



- [15] DLOUHÝ, M.; FÁBRY, J.; KUNCOVÁ, M.. *Simulace pro ekonomy*. Praha : VŠE, 2005.
- [16] NEPOR, Vladimír. *Rychlý průvodce simulacemi a simulačním softwarem*. Online. Dostupné z: <https://www.vrapps.cz/blog/simulace-a-simulacni-software>. [cit. 2024-04-23].
- [17] *RockwellAutomation*. Online. Dostupné z: <https://www.rockwellautomation.com/en-us.html>. [cit. 2024-04-23].
- [18] *Siemens*. Online. Dostupné z: <https://plm.sw.siemens.com/en-US/tecnomatix/products/plant-simulation-software/>. [cit. 2024-04-23].
- [19] *FlexSim*. Online. Dostupné z: <https://www.flexsim.com/>. [cit. 2024-04-23].
- [20] *VisualComponents*. Online. Dostupné z: <https://www.visualcomponents.com/>. [cit. 2024-04-23].
- [21] SHARMA, Prateek. Discrete-Event Simulation. *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*. 2015, roč. 4, č. 4, s. 5. ISSN 2277-8616.
- [22] SCHOLZ, Pavel. Počítačová simulace ve výrobě – nepublikované přednášky, Praha 2023 [cit. 2024-04-23]
- [23] KOCHENDERFER, Mykel J. a WHEELER, Tim A. *Algorithms for optimization*. Cambridge, Massachusetts: The MIT Press, [2019]. ISBN 978-026-2039-420.
- [24] PIDD, Michael. *Computer Simulation in Management Science*. 5th. John Wiley&Sons, 2004. ISBN 0-470-09230-0.
- [25] RICHARDS, Gwynne. *Warehouse Management: A Complete Guide To Improving Efficiency And Minimizing Costs In The Modern Warehouse*. 2nd. KoganPage, 2014. ISBN 978-0-7494-6934-4.
- [26] *Breezetre*. Online. Dostupné z: <https://www.breezetre.com/articles/spaghetti-diagram>. [cit. 2024-05-23].
- [27] *TheDataSchool*. Online. Dostupné z: <https://www.thedataschool.com.au/guillermo-correa/when-to-use-a-sankey-diagram/>. [cit. 2024-05-23].
- [28] *IKvalita*. Online. Dostupné z: <http://www.ikvalita.cz/tools.php?ID=25>. [cit. 2024-05-23].
- [29] *Wevalgo*. Online. Dostupné z: <https://www.wevalgo.com/know-how/lean-management/lean-methods-tools/vsm>. [cit. 2024-05-23].
- [30] *Daifuku*. Online. Dostupné z: <https://www.daifuku.com/solution/intralogistics/products/automated-warehouse/unitload-asrs/>. [cit. 2024-05-23].



- [31] *Researchgate*. Online. Dostupné z: [https://www.researchgate.net/figure/The-Sketch-of-the-Studied-3D-Compact-AS-RS\\_fig1\\_228287796](https://www.researchgate.net/figure/The-Sketch-of-the-Studied-3D-Compact-AS-RS_fig1_228287796). [cit. 2024-05-23].
- [32] *NSR Libya*. Online. Dostupné z: <https://nsr-libya.ly/>. [cit. 2024-05-23].
- [33] *Researchgate*. Online. Dostupné z: [https://www.researchgate.net/figure/A-shuttle-based-storage-and-retrieval-system-SBS-RS-from-a-single-aisle-view\\_fig1\\_350003445](https://www.researchgate.net/figure/A-shuttle-based-storage-and-retrieval-system-SBS-RS-from-a-single-aisle-view_fig1_350003445). [cit. 2024-05-23].
- [34] *Murderousmaths*. Online. Dostupné z: <http://www.murderousmaths.co.uk/games/loyd/loydfr.htm?moves=0>. [cit. 2024-05-23].
- [35] *Strongpoint*. Online. Dostupné z: <https://www.strongpoint.com/blog/automated-storage-and-retrieval-systems-asrs/>. [cit. 2024-05-23].
- [36] *Warehousenews*. Online. Dostupné z: <https://warehousenews.co.uk/2017/02/swisslogs-carrypick-db-schenker-logistics-the-e-commerce-race/>. [cit. 2024-05-23].
- [37] *Safran Group*. Online. Dostupné z: <https://www.safran-group.com/>. [cit. 2024-05-23].



# Seznam příloh

1x CD