

ZÁPADOČESKÁ UNIVERZITA V PLZNI

FAKULTA EKONOMICKÁ

Diplomová práce

Studie implementace logistického nástroje

Logistics tool implementation study

Richard Vaněk

Plzeň 2020

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Richard VANĚK**

Osobní číslo: **K17N0114P**

Studijní program: **N6209 Systémové inženýrství a informatika**

Studijní obor: **Systémy projektového řízení**

Název tématu: **Studie implementace logistického nástroje**

Zadávací katedra: **Katedra podnikové ekonomiky a managementu**

Z á s a d y p r o v y p r a c o v á n í :

1. Představte podnikatelský subjekt.
2. Popište nástroj PFEP.
3. Vypracujte návrh postupu implementace PFEP.
4. Provedte ekonomické zhodnocení implementace nástroje PFEP.

Rozsah grafických prací: **neuveden**
Rozsah kvalifikační práce: **60 - 80 stran**
Forma zpracování diplomové práce: **tištěná/elektronická**
Seznam odborné literatury:

- **CONRAD, Tim, ROOKS, Robyn.** *Turbo flow: using plan for every part (PFEP) to turbo charge your supply chain.* Boca Raton: CRC Press, 2011. ISBN 9781439820674.
- **FREDERICK, Ross David.** *Introduction to supply chain management technologies.* 2nd ed. Boca Raton: CRC Press/Taylor & Francis, 2011. ISBN 9781439837535.
- **HARRIS, Rick, HARRIS, Chris, WILSON, Earl.** *Making materials flow: a lean material-handling guide for operations, production-control, and engineering professionals.* Brookline, MA: Lean Enterprise Institute, 2003. ISBN 0974182494.
- **SMALLEY, Art.** *Creating level pull: a lean production-system improvement guide for production-control, operations, and engineering professionals.* Cambridge, MA: Lean Enterprises Institute, c2003. ISBN 0-9743225-0-4.
- **WAGNER, Bernd, ENZLER, Stefan.** *Material flow management: Improving cost efficiency and environmental performance.* Heidelberg: Physica-Verlag; 2006. ISBN 978-3-7908-1665-5.

Vedoucí diplomové práce: **Ing. Martin Januška, Ph.D.**
Katedra podnikové ekonomiky a managementu

Datum zadání diplomové práce: **23. října 2018**
Termín odevzdání diplomové práce: **23. dubna 2019**



Doc. Ing. Michaela Krechovská, Ph.D.
děkanka



Doc. PaedDr. Dana Egerová, Ph.D.
vedoucí katedry

V Plzni dne 23. října 2018

Čestné prohlášení

Prohlašuji, že jsem diplomovou práci na téma

„Studie implementace logistického nástroje“

vypracoval samostatně pod odborným dohledem vedoucího diplomové práce za použití pramenů uvedených v příložené bibliografii.

V dne

Podpis autora

Poděkování

Tímto bych chtěl poděkovat Ing. Martinovi Januškoví, Ph.D. za jeho cenné rady, vstřícný přístup a odborné vedení mé práce.

Dále bych chtěl poděkovat panu Ing. Petrovi Vašíčkovi a panu Ing. Petrovi Pivkovi ze společnosti WITTE Automotive (Witte) za poskytnutí podkladů a cenných rad.

Richard Vaněk

Obsah

Úvod	7
1 Představení WITTE Automotive	8
1.1 Základní informace	8
1.2 Výrobky Witte Automotive	9
1.2.1 Koncepty sedadel	9
1.2.2 Koncepty pro kapoty	10
1.2.3 Koncepty dveří	11
1.2.4 WITOL®	12
1.2.5 Flinkey	12
1.3 Řízení a optimalizace zásob ve Witte Automotive	13
1.3.1 Řízení zásob	15
1.3.2 Optimalizace zásob	17
2 Popis nástroje PFEP	23
2.1 Historie	23
2.2 Motivace	24
2.3 Představení nástroje	25
2.3.1 Přístupy k výstavbě PFEP	27
2.3.2 Návrh nástroje	29
3 Postup implementace PFEP	46
3.1 Kroky implementace	46
3.1.1 Předpoklady	46
3.1.2 Kroky tvorby nástroje	47
3.1.3 Management PFEP	50
4 Ekonomické zhodnocení zavedení PFEP	51
4.0.1 Rozbor skriptu simulace	52
4.0.2 Výsledek simulace	57
4.0.3 Porovnání variant řešení PFEP	63
Závěr	69
Seznam použité literatury	74
Seznam obrázků	76
Seznam tabulek	77
Slovník	78
Seznam zkratk	80
Seznam příloh	82
Absrakt	
Abstract	

Úvod

Nedílnou součástí provozu každé výrobní společnosti jsou zásoby. Spolu se zásobami přichází také náklady na jejich přepravu, interní manipulaci, jejich skladování, případnou likvidaci u výběhu starých dílů atd. Snížením nákladů firmy může být docíleno právě optimalizací a snížením skladových zásob na co možná nejmenší úroveň. Jedné se tedy o jakési „zeštíhlení“ firmy a skladů.

S podobnou myšlenkou musela pracovat už Toyota Motor Corporation (Toyota) po 2. světové válce, která nebyla preferována japonskou vládou a byla nucena hledat příležitosti pro úspory. Za těchto nelehkých podmínek vznikl Toyota production system (TPS), kterým se inspiruje v dnešní době nespočet firem na celém světě. Jednou z těchto firem je WITTE Automotive (Witte) zaměřující se na klíčové koncepty pro automobilový průmysl. Historie a oblast podnikání této firmy je popsána v první části diplomové práce. Úvodní text o firmě je následován popisem řízení a optimalizace zásob uvnitř firmy. Je zde také zmínka o hlavním Key Performance Indicator (KPI) zásob pro celou Witte group, kterým je Reichweite (RWT).

Z těchto dat poté vychází návrh struktury nástroje Plan for every part (PFEP), který si klade za cíl zefektivnit práci s tímto ukazatelem a poukázat na potenciál nástroje PFEP v dalších oblastech. Pro řešení problematiky simulace je použit řešitel úloh matematického programování. Druhá oblast výpočtů nástroje se zaměřuje na optimalizaci počtu kusů materiálu v obalu za pomoci Visual Basic for Applications (VBA) skriptu.

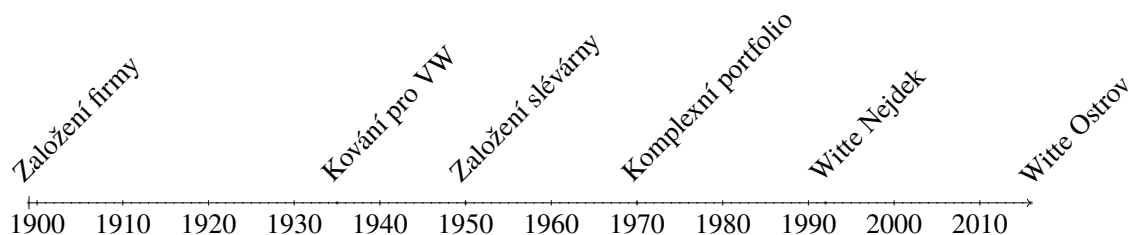
Konkrétní návrh nástroje je následován popisem obecného postupu při konstrukci nástroje PFEP ve firmách. Je zde kladen důraz na nutné kroky pro splnění vytyčeného cíle a není zde opomíjen ani klíčový faktor manažera PFEP. Finální část práce se zabývá simulací efektu implementace PFEP, která je vypracována v prostředí skriptovacího jazyka Python. Výstupy ze simulace jsou konfrontovány s náklady na zavedení interního Excel PFEP nástroje a komerčního řešení. Finální ekonomické zhodnocení zodpoví otázku hodnoty na vynaložené náklady pro 1 CZK úspor.

1. Představení WITTE Automotive

1.1 Základní informace

Prvopočátky působení firmy WITTE Automotive (Witte) sahají do roku 1899. V tomto roce zakládá pan Ewald Witte v německém Velbertu firmu zaměřující se na výrobu zámků na kufry, rozličných předmětů do domácnosti a kuchyně. Vstup na automobilový trh nastal před vypuknutím 2.světové války v podobě kování pro vůz VW „Brouk“. Úspěchy firmy podpořila i vzniklá slévárna v 50.letech minulého století, která umožnila rozšířit portfolio nabízených výrobků pro automobilový průmysl. V 70. letech firma rozšířila sortiment na všechny zamykací systémy automobilů.

Obrázek 1.1: Historie Witte



Zdroj: Vlastní zpracování: data z

<https://www.witte-automotive.cz/live/Historie/historie.aspx>

V roce 1992 je založeno Witte Nejdek v České republice a firma se takto stává významným hráčem v automobilovém průmyslu. Svou pozici dále upevňuje v roce 1993 dohodou s tchajwanskou firmou, kterou pak dále rozšiřuje o zapojení do aliance VAST¹, ADAC Automotive² a STRATTEC Security Corporation³.

Rozšiřování působení firmy pokračovalo dále zahájením výroby v Bulharsku v roce 2010 a postavením vlastního závodu v bulharském Ruse v roce 2014. A konečně v roce 2016 otvírá Witte svojí pobočku v Ostrově, která mimo jiné disponuje i plně automatickou lakovací stanicí o dvanácti robotech.

[18, 35, 39]

¹Mezinárodní aliance výrobců zamykacích a bezpečnostních systémů pro automobilový průmysl.[36]

²Americký výrobce automobilových dílů.[17]

³Americká společnost zabývající se bezpečnostními a zamykacími systémy do aut.[16]

- **Datum vzniku a zápisu:** 11. prosince 1991
- **Obchodní firma:** WITTE Nejdek, spol. s r.o.
- **Sídlo:** Rooseveltova 1299, 362 21 Nejdek
- **Identifikační číslo:** 40525881
- **Právní forma:** Společnost s ručením omezeným
- **Základní kapitál:** 97 000 000,- Kč
- **www:** <https://www.witte-automotive.cz/>

[37]

1.2 Výrobky Witte Automotive

V současné době se Witte primárně zaměřuje na automobilový trh. Portfolio výrobků lze rozdělit na tyto hlavní kategorie:

1. Koncepty sedadel
2. Koncepty pro kapoty
3. Koncepty dveří
4. WITOL®
5. Flinkey

1.2.1 Koncepty sedadel

Witte provedlo rozsáhlé studie bezpečnosti sedadel a díky tomu mohlo vyvinout zámky opěradel s garancí vysoké ochrany cestujících. Pomocí zámků opěradel a zamykacích třmenů absorbujících energii, je při nárazu minimalizované zatížení na opěradla. Dostupné jsou rovněž speciality na přání v podobě sedačkových zámků s elektronickou nebo mechanickou signalizační funkcí a schopnosti propojit vzájemně bezpečnostní pásy. Do této kategorie můžeme zařadit:

- Sedačkové zámky

Obrázek 1.2: Výrobky Witte



Zdroj: <https://www.witte-automotive.cz/>

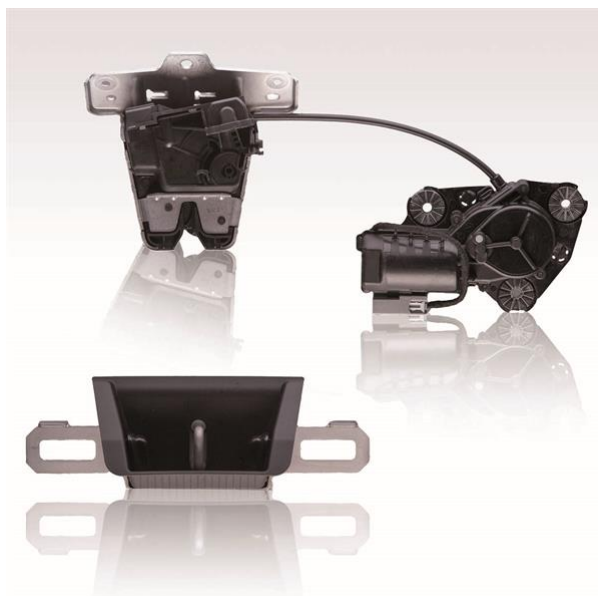
- Háky
- Vnitřní kliky
- Pohony
- Zámky pro fixaci sedadla k podlaze
- Zámky opěradel s multipozičním nastavením atd.

1.2.2 Koncepty pro kapoty

Witte poskytuje komplexní nabídku produktů pro zád' vozu. Výrobky v této oblasti jsou například:

- Kamerové systémy
- Kapotové zámky
- Osvětlení SPZ
- Zámky zadního okna
- Systém ochrany chodců atd.

Obrázek 1.3: Výrobky z řady kapotových konceptů



Zdroj: <https://www.witte-automotive.cz/live/kapoty/kapoty.aspx>

1.2.3 Koncepty dveří

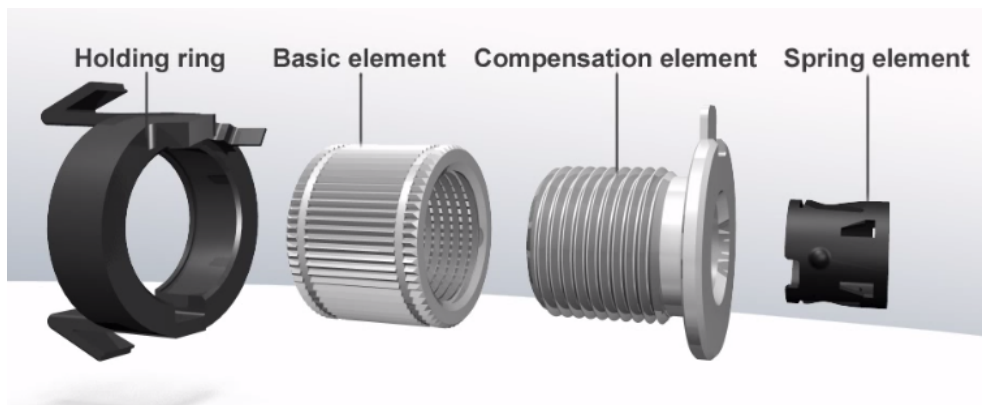
Firma dokáže využít potenciál pracovníků výzkumu a vývoje a na trhu může díky svému patentovanému západkovému systému nabídnout zámky bezpečné, ale i s menší rozměrovou stopou. Moderní trend odemknutí a zamknutí automobilu při odchodu z jeho blízkosti je díky Witte také možný prostřednictvím systému Passive Entry / Passive Go. K zefektivnění výroby a zkvalitnění nabízených klik posloužila výstavba a otevření nové ostrovské výrobní haly v roce 2016. Díky této hale je Witte schopno dodávat kliky v potřebné kvalitě a korektním zákaznickem požadovaném odstínu. V oblasti konceptů dveří se můžeme dále bavit o:

- Dveřní moduly
- Výztuhy dveří
- Panty
- Zamykací čepy
- Brzdy
- Passive Go
- Ochrana hrany dveří

1.2.4 WITOL®

Produkt přesahující svou aplikací automobilový průmysl, to je systém WITOL®. Jedná se o automatický systém vyrovnávání tolerancí při montáži různých komponent. Skládá se z prstence, základního elementu, kompenzačního elementu a pružinky. Struktura systému je patrná na obrázku 1.4 na straně 12.

Obrázek 1.4: Systém WITOL®



Zdroj: <https://www.witol.com/>

1.2.5 Flinkey

Flinkey je plug and play řešení pro bezklíčovou obsluhu auta. Řešení umožňuje spravovat přístup k vozidlu přes mobilní aplikaci a tímto například umožnit fungování car-sharingových služeb jako je Karkulka, Anytimecar, Uniqueway a další.

[38]

Globální trh

Firma dodává své výrobky nejen výrobcům osobních automobilů, mezi které patří například Toyota Motor Corporation, Volvo Cars, Škoda Auto, Dodge, Ferrari, Automobili Lamborghini S.p.A., Ford Motor Company, ale také výrobcům komerčních vozidel (Volvo Trucks, DAF Trucks, Scania atd.) a dodavatelům systémů (Valeo, Brose Fahrzeugteile GmbH Co. KG, Faurecia atd.)

Výrobci komerčních vozidel: Iveco, MAN, Renault Trucks, Scania Trucks, Volvo Trucks atd.

Systémoví dodavatelé: Lear corporation, Bosch, Faurecia, Magna, Valeo, Brose, Visteon, Johnson Controls atd. [27]

1.3 Řízení a optimalizace zásob ve Witte Automotive

Globální propojení lidí a trhů představuje pro firmy příležitost, ale zejména tlak a konkurenční boj, který panuje nejen na automobilovém trhu, ale i v dalších oblastech podnikání. Zákazníci si mohou vybírat z celé řady produktů ve velmi podobné kvalitě. Pro firmy to znamená boj o uspokojení potřeb zákazníka lépe než konkurence, a tímto i ukrojení si z pomyslného koláče trhu.

Zákaznický relevantní kritéria pro rozhodování o dodavateli jsou například **excelentní zákaznický servis**, jenž je velmi dobře znám a používán například v bankovním sektoru. Dále si dnes zákazníci potrpí na **delší termíny splatnosti faktur**. A v neposlední řadě se zákazníci zaměřují na **cenu poptávaných produktů**. Firma musí nalézt cestu ke snížení ceny svých nabízených výrobků při zachování či navýšení zisku. Je tedy nutné snižovat své náklady v různých oblastech svého fungování. Jednou z těchto oblastí jsou zásoby, které představují nemalou položku vázaných finančních prostředků a zpravidla se pohybují od 10% do 25% celkových aktiv podniku. Z tohoto důvodu může mít i nepatrné snížení zásob značný vliv na podnik.

Graf 1.5 zobrazuje reálný pohled podílu zásob na celkových aktivech Witte v letech 2006 až 2018⁴. Z grafu je patrný oscilující trend kolem průměru 18% s minimy 16% v letech 2007, 2009, 2015 a 2016. Zásoby vstupují do hospodaření podniku prostřednictvím logistických nákladů, které můžeme rozdělit do následujících kategorií:

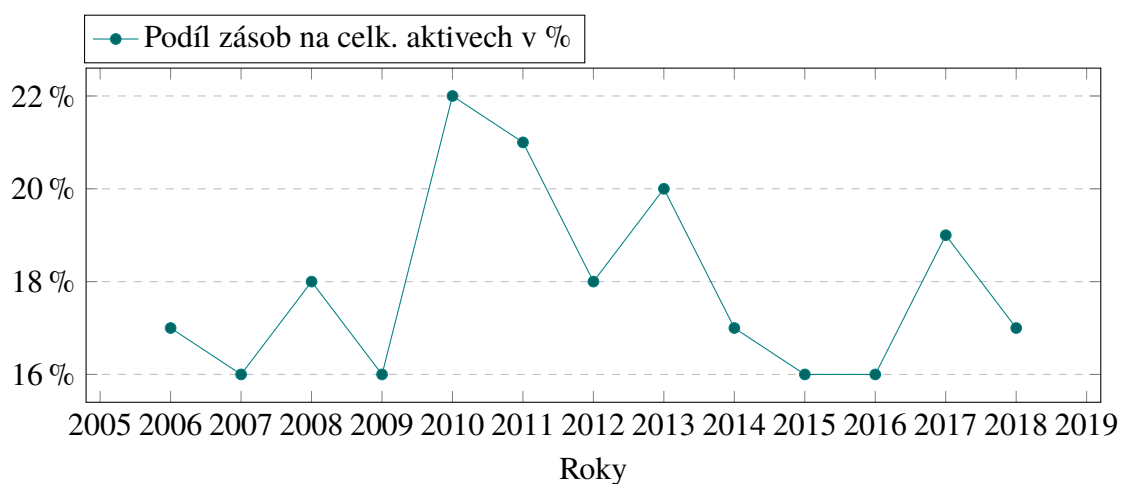
1. Skladovací prostory - Vlastní / externí (Vícenáklady).
2. Vázaný kapitál - Náklady na úroky z financování kapitálu vázaného v zásobách. Rozdíly v termínech splatnosti od zákazníků a dodavatelů, termíny investic atd. znamenají nutnost využít krátkodobých úvěrů - kontokorent.⁵

⁴Zdrojová data převzata z rozvahy společnosti za roky 2006-2018. Výsledná hodnota je podíl zásob a celkové hodnoty aktiv.

⁵Povolené čerpání finančních prostředků pod zůstatek na účtu klienta. Klientovi nabíhají úroky za každý den, který je v mínusu.

3. Obalový materiál - obaly jsou použity na materiál ve skladu a chybí v jiných částech firmy. Nutné řešit dodatečným nákupem nebo pronájemem obalového materiálu.
4. Manipulace - podmínka dodržení First in first out (FIFO) má za následek zvýšení manipulace při větším objemu materiálu.
5. Doprava.
6. Šrotace - Nekorektně řízený náběh a výběh materiálu. Náhlá změna potřeb zákazníka.

Obrázek 1.5: Podíl zásob na celkových aktivech Witte



Zdroj: Vlastní zpracování, data:

<https://or.justice.cz/ias/ui/vypis-sl-firma?subjektId=151117>

Logistické náklady jsou jednou z komponent nákladové a cenové kalkulace výrobků Witte. Ze schématu kalkulace na obrázku 1.6 můžeme odvodit následující rovnice výpočtu celkové ceny výrobku:

$$Cena = UVNV + ZP \quad (1.1)$$

$$UVNV = VNV + SOR \quad (1.2)$$

$$VNV = MN + VN \quad (1.3)$$

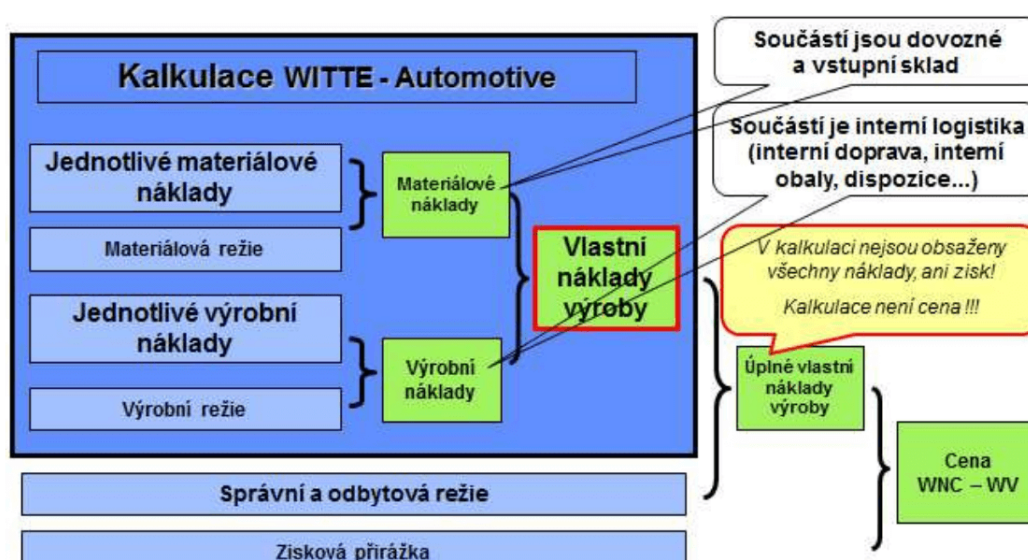
$$MN = jmn + mr \quad (1.4)$$

$$VN = jvn + vr \quad (1.5)$$

kde:

jmn = Jednotkové⁶ materiálové náklady, **mr** = Materiálová režie, **MN** = Materiálové náklady, **jvn** = Jednotkové⁷ výrobní náklady, **vr** = Výrobní režie, **VN** = Výrobní náklady, **VNV** = Vlastní náklady výroby, **UVNV** = Úplné vlastní náklady výroby, **SOR** = Správní a odbytová režie, **ZP** = Zisková přírážka

Obrázek 1.6: Logistické náklady v kalkulacích



Zdroj: Interní materiály Witte[6]

1.3.1 Řízení zásob

Důležitost Supply chain managementu a logistiky si firma Witte uvědomuje a dlouhodobě usiluje o zavádění rozličných „štíhlých“⁸ praktik do svých interních procesů. Nejdecký závod například zahájil v roce 2009 pilotní projekt ve Witte group na systém zásobování „Milk run“, který je nyní nezbytnou součástí běhu nejen pobočky v Nejdku, ale například i závodu v Ostrově. Mimo zmíněného systému zásobování používají pracovníci plánování

⁶ „Jednotlivé“ ve zdrojovém materiálu

⁷ „Jednotlivé“ ve zdrojovém materiálu

⁸ Štíhlá výroba je metodika vyvinuta firmou Toyota po 2. světové válce. Cílem tohoto přístupu je vyhovět zákazníkovi v maximální možné míře a za co nejmenších nákladů a plýtvání.

logistiky k řízení zásob několik nástrojů, jako jsou například pojistné zásoby, karenční dny atd.[40, 25, 24, 26]

Pojistné zásoby

Pojistnou zásobu zadává do systému SAP příslušný disponent. Informace se uloží pouze při zadání termínů trvání a důvodu pojistné zásoby. Výše pojistné zásoby představuje trvalou nadzásobu materiálu, která vede ke kapitálové vázanosti v materiálu, zvýšenému objemu obalového materiálu, potřebě skladovacích míst a navýšení nároků na manipulaci.

Pojistná zásoba dle interních statistik „stojí“ Witte více jak 20% hodnoty této zásoby. Interní metodika doporučuje používat spíše karenční dny. V případě užití pojistné zásoby je nutné pravidelně hodnotit a kontrolovat důvody pro nastavení výše pojistné zásoby. [4, 13, 5]

Karenční dny

Karenční dny jsou parametrem, který slouží k předsunutí požadavků dodávky o zadaný počet dní. SAP ve svém běžném nastavení počítá s kompletním vyčerpáním zásob ve skladu před další dodávkou. Předsunutím požadavku může pracovník vytvořit „pojistnou“ zásobu sloužící k pokrytí výkyvů ve výrobním procesu. Výhoda v porovnání s pojistnou zásobou je nenavýšení materiálu ve skladu nad požadavky zákazníka, ale zákaznické požadavky jsou pouze předsouvány.[5]

Profil doby dosahu

Jedná se o dynamickou pojistnou zásobu, která je rozdělena do několik úseků řídicí se jednotlivými dobami dosahu zásob. Výhodou tohoto nástroje je nenavyšování skladových zásob, ale pouze přerozdělení potřeb do časových úseků. Profil dosahu je vypočítáván z průměrných denních potřeb, která se počítají z nastaveného profilu. [5]

LSF

LSF slouží k vytvoření mimořádného požadavku na materiál bez ohledu na skladové zásoby. Jeho použití je nutno velmi dobře promyslet a zvolit jej pouze, když není možné použít jinou možnost. Důvody pro použití LSF mohou být:

- Inventurní rozdíl
- Chybějící požadavky od zákazníka
- Chybějící požadavky dodavatele
- Nařízení nákupu
- Tvorba předzásob
- Start nového projektu
- Objednání vzorků

[5]

1.3.2 Optimalizace zásob

Pro splnění firemních cílů být konkurenceschopný, zvýšit zákaznickou satisfakci a navýšit zisk, je nutné provádět neustálá zlepšování a optimalizace skladových zásob. Firma musí provést řadu úkonů, které povedou ke snížení logistických nákladů. Jedná se například o nastavování a snahu o minimalizaci pojistných zásob a karenčních dnů, minimalizace LSF požadavků, změna balení atd. [6]

Reichweite (RWT)

Ukazatel dosahu zásob s ohledem na budoucí požadavky na úrovni celé Witte Group. Požadavky v minulosti nejsou zohledňovány. Jedná se také o jeden z Key Performance Indicator (KPI), který má stanovenou konkrétní hodnotu pro každý závod, případně Business Team (BT). V roce 2007 byl pro Witte Nejdek cíl RWT 5,0, pro rok 2008 pak 4,9.

RWT je důležitý a používaný indikátor, na který je možno nahlížet v různých úrovních a pohledech:

- Za celou Group, závod, BT, disponenta
- Podle třídy ocenění, statusu materiálu, ABC
- Podle období - ročně, měsíčně, týdně, denně

Výpočet RWT je následující

$$Reichweite = \frac{hz}{\frac{hp}{20}} \quad (1.6)$$

kde: **hz** = Hodnota zásob, **hp** = Hodnota požadavků na následujících 20 dnů

Příklad výpočtu: Na skladu máme 2000ks materiálu X s cenou 10€ za 1ks, hodnota zásob = 20 000€. Suma požadavků na následujících 20 dnů je 5000ks X, hodnota požadavků = 100 000€.

$$ReichweiteX = \frac{20000}{\frac{100000}{20}} \quad (1.7)$$

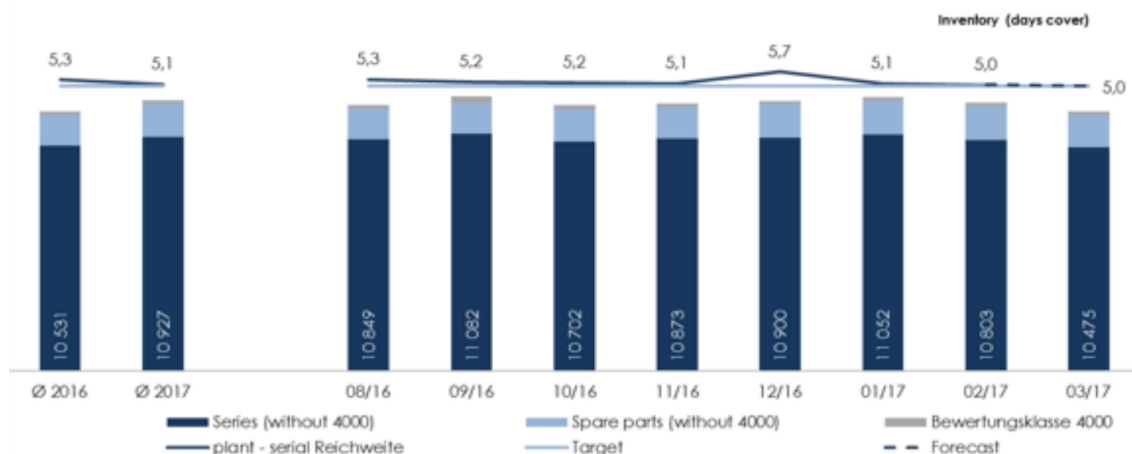
$$ReichweiteX = 4 \quad (1.8)$$

[6, 4]

Reporting RWT

Důležitost KPI RWT umocňují měsíční schůzky s vedením logistiky Witte Group, probíhající každý první pátek v měsíci. Přes telekonference je probírán vývoj RWT, případná

Obrázek 1.7: Příklad měsíčního reportingu RWT



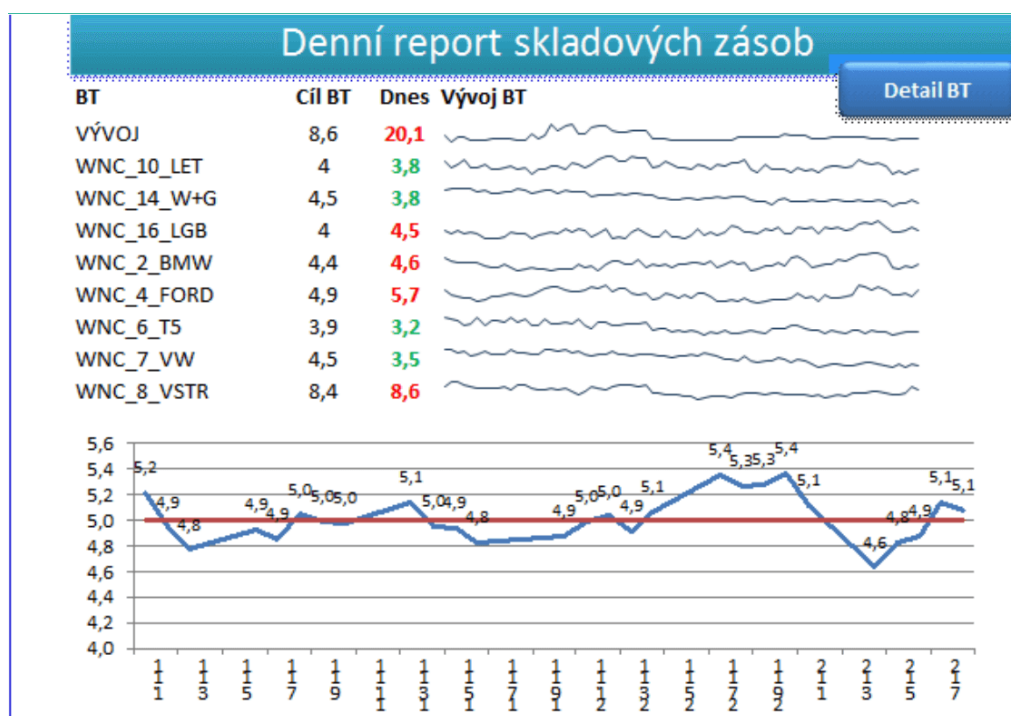
Zdroj: Witte interní zdroj

opatření k nejhlavnějším položkám, které se dělí na sériové (Tmavě modrý sloupec grafu,

tvoří většinu zásob), náhradní (Světle modrý sloupec grafu) a vývojové díly (Šedý sloupec grafu). Příklad reportu je zobrazen na obrázku 1.7. Vidíme zde sloupce zásob a dvě indikační čáry. Tmavě modrá z těchto čar zobrazuje reálný obraz RWT sériových dílů. Světle modrá poté znázorňuje plán RWT pro tyto díly. Přerušovaná je předpovídaná hodnota RWT.

Dále je používaný týdenní a denní RWT reporting. Příklad denního reportingu je na obrázku 1.8. Je zde vidět rozpad údajů pro jednotlivé BT, cíle BT pro RWT, aktuální stav a vývoj RWT v čase. Tento report zasílá odpovědný pracovník SCM vedení logistiky, dispozici a vedení BT.

Obrázek 1.8: Příklad denního reportingu RWT



Zdroj: Witte interní zdroj

Simulace RWT

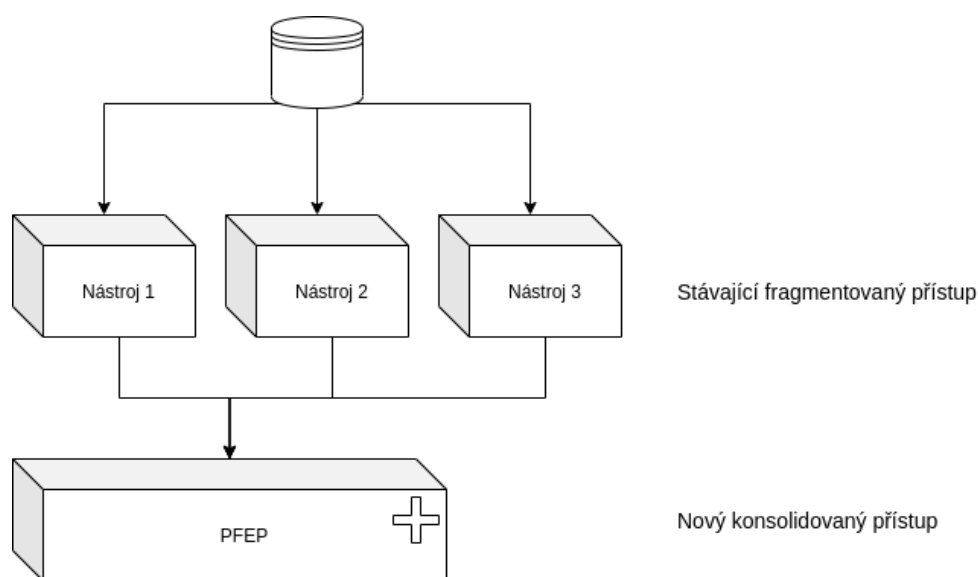
Pracovníci SCM Witte mají pro svou práci k dispozici simulační nástroj v MS Excel, pro který získávají data ze SAP Business warehouse (BW) portálu. BW portál slouží k shromáždění dat ze systému a jejich následný export.

Lze zde ukládat i předdefinované reporty pro pohodlí uživatelů. Potřeby pracovníků pokrývají následující základní reporty RWT:

1. Reichweite pro Werk von bis Tabelle - základní a nejčastěji používané zobrazení sloužící pro simulace a analýzu TOP položek
2. Reichweite pro Werk und Jahr Tabelle - zobrazení RWT za vybraný rok
3. Reichweite pro Werk Itz. 12 Monat Grapik - zobrazení za posledních 12 měsíců
4. Reichweite pro Werk Itz. 10 Wochen Grapik - zobrazení za posledních 10 týdnů
5. Reichweite pro Werk Itz. 30 Wochen Grapik - zobrazení za posledních 30 týdnů
6. Reichweite pro Werk Itz. 30 Tage Grapik - zobrazení za posledních 30 dnů

Samotný simulační nástroj je vytvořený v Excelu a každý BT má uloženou svojí verzi na sdíleném disku. Výstupní data z BW načte pracovník do Excelu, seřadí díly podle hodnoty průměrné skladové zásoby, aktualizuje kontingenční tabulku a tímto má připravené prostředí pro práci s nástrojem. Více tento proces popisuje diagram 1.10 na stránce 22.

Obrázek 1.9: Změna přístupu v SCM řízení

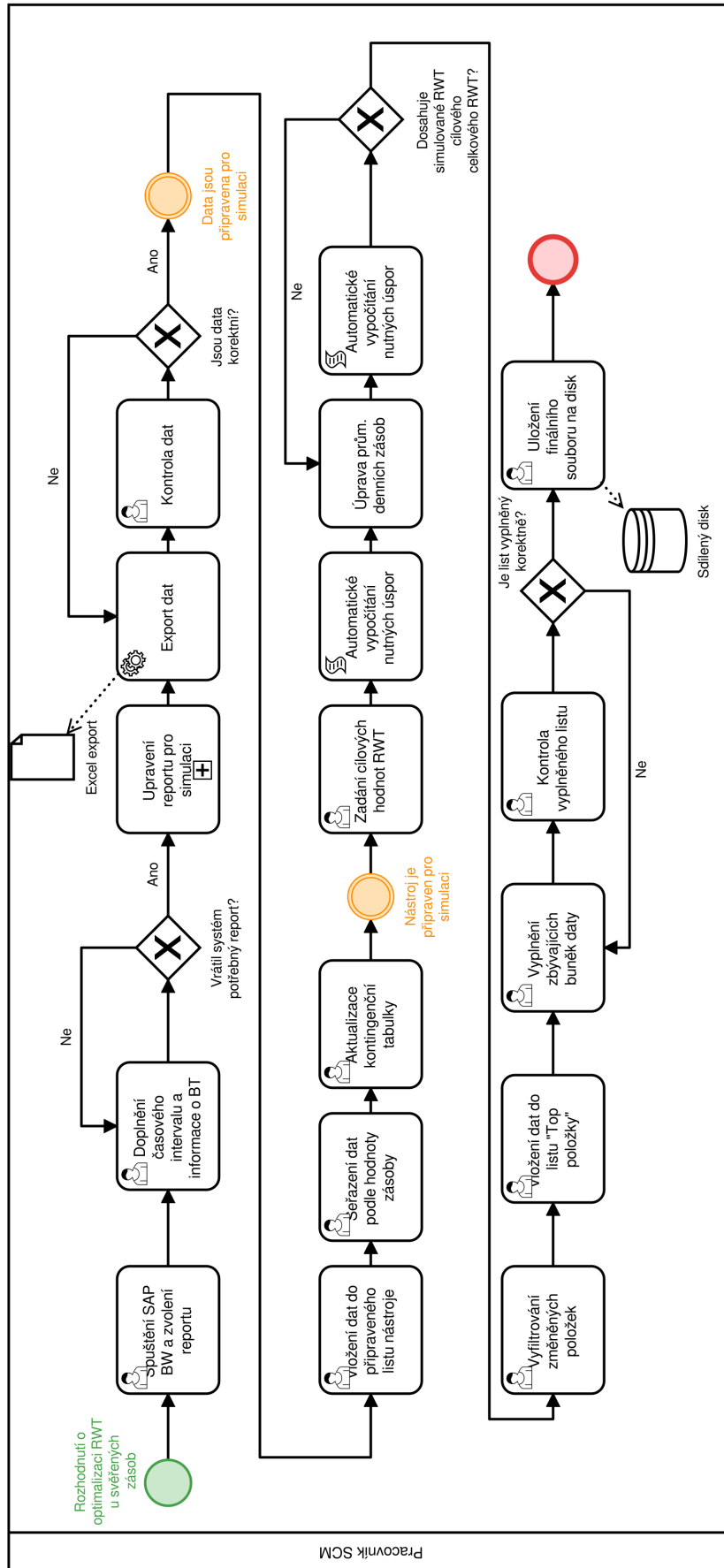


Zdroj: Zdroj: Vlastní zpracování

Hodnotu RWT ovlivňují mimo výše hodnoty zásob i jiné faktory. Jedná se například o čištění COGI, pojistné zásoby, záporné stavy, balící předpisy, výběhy, náběhy, minimální odběrové množství atd. Analýza a optimalizace RWT však neposkytuje komplexní pohled na dopad změny proměnných jednotlivých materiálových položek. Nedozevíme se například dopad změny objednávaného množství na skladové prostory či na počet kolujících interních obalů.

Tento nedostatek globálního pohledu na skladové hospodářství nabádá k jednotnému nástroji, který bude obsahovat veškerá potřebná data a dokáže poskytnout pracovníkovi užitečné informace pokrývající větší spektrum SCM. Musí nastat posun od štěpení analýz na fragmenty ke konsolidaci dat a provádění simulací v tomto prostředí zahrnujícím více proměnných pro každou materiálovou položku. Nástroj musí představovat ucelený systém zahrnující datovou komponentu, jež bude zásobovat reportovací a simulační komponentu. Plánování na úrovni jednotlivých položek, Plan for every part (PFEP).

Obrázek 1.10: Proces simulace optimalizace RWT



Zdroj: Vlastní zpracování dle interních materiálů Witte

2. Popis nástroje PFEP

2.1 Historie

Historie PFEP sahá do počátků vzniku Lean manufacturing myšlení ve společnosti Toyota (Dříve Toyoda Loom Works) v Japonsku, která se nacházela v nelehké situaci po skončení 2. světové války. Jistý odbyt za 2. světové války, který byl tvořen zejména poptávkou nákladních vozů japonskou vládou opadl po jejím konci. Japonská vláda následně představila velmi strukturované ekonomické plánování, jež poskytovalo firmám finanční prostředky pouze za schválení Ministerstva obchodu a průmyslu MITI. Toyota byla shledána nadbytečnou firmou v oblasti produkce automobilů, Mitsubishi a Nissan zvládly dle MITI pokrýt japonskou poptávku.

Toyota s tímto krokem nesouhlasila a i přes nezískání finančních prostředků na modernizaci od japonské vlády pokračovala v produkci automobilů, které hodlala nabízet na globálním trhu. Toto omezení nutilo firmu se zamyslet nad svými procesy a přijít na způsob umožňující prodej auta v první řadě, obdržení finančních prostředků od zákazníka a až po dokončení finanční transakce zahájit výrobu požadovaného kusu. Taiichi Ohno¹ a Shigeo Shingo² čerpali inspiraci od západních firem a to konkrétně od Ford Motor Company (Ford). Získané poznatky nenechali ležet ladem, ale postupně je začleňovali do svého myšlení a postupně tvořilo dnešní známý Toyota production system (TPS). Došli k závěru, že zásoby hrají významnou roli v běhu firmy, a proto je nutné se na ně cíleně zaměřit. Fordovy procesy se pro správné fungování v prostředí Toyoty musely modifikovat a být Just in time. Znamenalo to radikální změnu dodavatelsko-odběratelských procesů.

Dále svou pozornost zaměřili na interní procesy s cílem být flexibilnější a minimalizovat skladové zásoby. Tohoto chtěli docílit produkcí várek s nízkým počtem kusů. Taiichi Ohno požádal svého kolegu Shigeo Shingo, aby zahájil práci na redukci doby přenastavení strojů, neboli Single Minute Exchange of Die (SMED). Výrazná redukce času pro nastavení strojů by znamenala konkurenční výhodu a flexibilitu oproti zmíněnému Fordu.

¹Byl to japonský inženýr a podnikatel. Je považován za zakladatele Toyota Production System.

²Byl to japonský inženýr. Kolega Taiichi Ohna a expert na Toyota Production System.

Toyota také nesouhlasila s americkým přístupem k zaměstnancům jako nemyslícím biologickým strojům. Namísto toho se snažili o kultivaci zaměstnanců a využití jejich znalostí a zkušeností v pracovním procesu pro představení zlepšovacích návrhů. Tento proces pojmenovali Kaizen.

Toyota vytvořila „Pull“ systém výroby, který produkuje podle požadavků zákazníka, nikoliv podle predikce výrobce. Ideální stav nulových skladových zásob nelze docílit z různých důvodů neobsáhnutí všech proměnných, které v reálném světě zasahují do fungování firemních procesů. Proto je nutné mít určité minimální zásoby finálních kusů výroby, které si může zákazník prohlédnout, na místě zakoupit a v případě automobilového průmyslu následně se svým vozem odjet. Vznik prostředníků a prodejců je tedy v reálném světě nutný. Omezené zdroje prostředníků tlačí opět na minimalizaci skladových zásob, které musí být dostatečné pro pokrytí rozličných potřeb zákazníků. [2, 8, 13]

2.2 Motivace

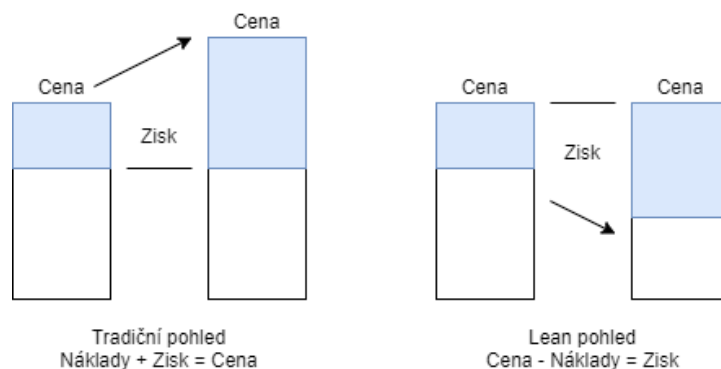
Tímto se dostáváme k samotnému nástroji PFEP. Jak již bylo zmíněno, Witte zásadním způsobem čerpá z TPS a používá Just in time pro své materiálové zásobování. Stejně tak jako Toyota a její prodejci se musí zamýšlet nad optimálním řízením skladových zásob.

Proto je nezbytnou součástí Lean manufacturing PFEP, který zde figuruje jako centrální bod pro zodpovězení otázek optimálního množství specifického materiálu, jaký produkt se bude daný den vyrábět a například může asistovat i při stavbě skladu či tvoření layoutů montážních stanic. PFEP je flexibilní a přizpůsobí se potřebám organizace. Cílem TPS a samotné organizace je eliminovat či omezit plýtvání. Tato myšlenka je otisknuta i v přístupu cenotvorby a pohledu na zisk ve firmě Toyota, která je znázorněna na obrázku 2.1 na straně 25. V klasickém pojetí cenotvorby se při zvýšení nákladů zvyšuje i cena výrobku za předpokladu zachování zisku. Lean pojetí vnímá zisk jako výsledek odečtení nákladů od ceny. K zachování stávajícího zisku musí firma nalézt opatření pro snížení svých nákladů.

Dále si TPS klade za cíl eliminovat sedm hlavních druhů plýtvání.

1. Nadvýroba - Výroba počtu kusů nad zákaznický požadavek. Výroba produktů z důvodu vytížení strojů. Výroba produktů, které se expedují příští týden.

Obrázek 2.1: Porovnání klasického pojetí cenotvorby s Lean pojetím



Zdroj: Vlastní zpracování dle [2]

2. Prostoje - Čekání na dodávku materiálu. Prostoje z důvodu seřizování a čištění strojů. Poruchy strojů.
3. Manipulace - Nesmyslná přeprava materiálu uvnitř firmy. Přesun materiálu z jedné části skladu do druhé. Transport ze skladu k montážní lince.
4. Technologické ztráty - Špatné technologické procesy. Kontrola materiálu. Práce na části dílu, která není ve výsledném produktu viditelná.
5. Zásoby - Nadbytečné zásoby materiálu, polotovarů a dokončených výrobků.
6. Pohyb - Pohyb, který nepřidává na hodnotě výrobku. Otáčení se pro výrobek, hledání dílu v přepravce, atd.
7. Opravy - Opravy materiálu, které neprošly kontrolou. Reklamacce materiálu, atd.

[2, 20]

2.3 Představení nástroje

Jak již bylo naznačeno v 1.3.2 na straně 21, PFEP je nástroj seskupující různá data z organizace do jednoho centrálního bodu. Data takto uložená slouží k tvorbě materiálových simulací pro pracovníky dané organizace.

Dle Conrada je PFEP „základním kamenem nejen pro určení hladiny zásob, ale také poskytuje nutné informace pro rozhodování o obalech, úložných regálech, frekvenci dodávek, plánování layoutů.“ [2, s. 8] Yang vidí tento nástroj jako „databázi pro každý díl ve firmě.“

Obsahuje detailní informace od nákupu až po výrobu, zahrnující číslo dílu, místo spotřeby, rozměry dílu, velikost obalu, počet kusů v obalu, informace o dodavateli a další.“[9, s. 3]

Samotný obsah databáze PFEP je přizpůsobený potřebám jednotlivých firem, které využijí výstupy tohoto nástroje pro různé účely. Podstata nástroje však zůstává totožná. Hlavním cílem je poskytnout validní a aktuální informace o jednotlivých komponentech ve firmě, ke kterým má každý pracovník snadný přístup. Pro získání většiny potřebných dat k tvorbě PFEP lze užít stávající datovou infrastrukturu firmy. Hlavním zdrojem dat bude ve většině podniků zejména Enterprise resource planning (ERP) systém firmy. Tabulka 2.1 na straně 26 nám zobrazuje informace o penetraci ERP ve firmách operujících na území ČR. Z tabulky je patrná značná penetrace těchto systémů ve středních a velkých podnicích.[28, 2]

Tabulka 2.1: Přehled podniků používající CRM a ERP systémy v ČR

	Počet podniků celkem	ERP v podnicích	CRM v podnicích
10–49 zaměstnanců	30 196	19,5%	14,7%
50–249 zaměstnanců	6 428	53,3%	34,3%
250 a více zaměstnanců	1 524	81,1%	41,9%
Podniky celkem (10+)	38 149	27,7%	19,1%

Zdroj: Vlastní zpracování, data: <https://www.systemonline.cz/erp/lokalni-i-globalni-trendy-erp-pro-rok-2015.htm>

ERP systémy

ERP systémy jsou jedním z nejkompexnějších a zároveň nejmocnějších podnikovým informačním systémem, který obhospodařuje mnoho základních firemních procesů. Nejedná se však pouze o databázi firemních dat, mnoho systémů je schopno automatizovat značnou část firemních procesů a tím ušetřit náklady. Typicky jsou ERP systémy využívány ve výrobě, finančním oddělení, obchodu, distribuci, lidských zdrojích atd.

Přehled top hráčů na trhu ERP systémů pro střední, velké a nadnárodní korporace je znázorněn na obrázku 2.2 na straně 27, který je vypracován společností Gartner³. Pro představu o pozici na trhu je SAP používaný společností Witte v roce 2019 umístěn v poli vizionářů především díky rozvoji svého produktu SAP S/4HANA⁴. [7, 19]

³Uznávaná výzkumná a poradenská společnost v oblastech IT, HR, financí atd.

⁴Nástupce SAP R/3 a SAP ERP.

Obrázek 2.2: Magický kvadrant ERP systémy - střední, velké a nadnárodní korporace



Zdroj: <https://www.gartner.com/doc/reprints?id=1-60FH5L0&ct=190514&elqTrackId=249a4412637c4cd19bc67e4b76a19fcb&elqaid=82783&elqat=2>

2.3.1 Přístupy k výstavbě PFEP

Samotná výstavba nástroje se odvíjí od objemu dat, složitosti a počtu výpočtů, schopnostech interních pracovníků a případně od finančních prostředků, které může firma alokovat na nákup externích služeb, když firma pociťuje absenci schopných interních pracovníků, kteří by tento úkol zvládli. Obecně se pro implementaci PFEP nabízí tři relevantní cesty. První je využití tabulkových kalkulátorů pro stavbu nástroje, při vyšší náročnosti a objemu dat je vhodné zvolit řešení na bázi relační databázi nebo lze zvolit jeden z dostupných komerčních softwarových řešení.

Tabulkové kalkulátory

Obecně lze definovat tabulkový kalkulátor jako počítačový program využívající dvoudimenzionální souřadnicové síť pro zobrazení, zachycení a práci s daty. Prvotně tabulkový kalkulátor sloužil zejména v oblasti účetnictví pro zápis finančních dat firem. Stavebním kamenem těchto velmi rozšířených nástrojů se stal v roce 1979 VisiCalc napsaný pro po-

čítač Apple II. Dnes již tabulkové kalkulátory dominují v mnoha oborech a jsou součástí počítače každého vědce, studenta a kancelářského pracovníka.

Právě tato rozšířenost zpřístupňuje tvorbu nástroje PFEP i pracovníkům nevybaveným programátorským umem. Prostředí těchto softwarových nástrojů je proto ideální pro nenáročné instance PFEP či rychlé prototypování funkcionality více propracovaného a robustnějšího nástroje založeného na relačních databázích.[32]

Databázové řešení

Relační databáze mají obdobnou strukturu ukládání dat jako tabulkové kalkulátory, na rozdíl od nich však neobsahují formátování, ale pouze data ve své surové podobě. Tato data jsou uložena v tabulkách, které sestávají ze sloupců obsahující vždy jen jeden datový typ. Řádky poté představují vždy jeden záznam popsany svými atributy, které jsou uloženy ve zmíněných sloupcích. Dále můžeme definovat vazby mezi tabulkami pomocí primárních a cizích klíčů.

Mezi nejčastější datové typy patří **číselné hodnoty** (binární hodnoty, celá čísla, reálná čísla atd.), **datum a čas**, **textové řetězce**, **unicode znaky** a případně další jako JSON či XML formáty. Hlavní výhodou oproti předchozímu softwarovému řešení v podobě tabulkových kalkulátorů je integrita dat, rychlost zobrazení dat, schopnost pojmuti objemnější soubor dat a snazší napojení na externí systémy.

Ke komunikaci a úpravě relačních databází je používán Structured Query Language (SQL). Jazyk se skládá z Data Definition Language (DDL), Data Control Language (DCL) a Data Manipulation Language (DML) částí. Syntaxe jazyka sestává z **klauzulí** (SELECT, FROM, WHERE), **predikátů** (name='Julie') a **výrazů**('Julie').

```
SELECT * FROM table1 WHERE name='Julie';
```

Mezi základní klauzule patří:

- SELECT - extrakce dat
- UPDATE - aktualizace dat
- DELETE - mazání dat

- INSERT INTO - vložení nových dat
- CREATE DATABASE - vytvoření databáze
- ALTER DATABASE - modifikace databáze
- CREATE TABLE - vytvoření nové tabulky
- ALTER TABLE - modifikace tabulky
- DROP TABLE - smazání tabulky

[33, 30, 34, 1]

Existující softwarová řešení

Poslední možností je využití již existujících komerčních řešení. Motivací k přijetí této volby bude zejména neproveditelnost tvorby nástroje v rámci firmy z důvodů absence personálu, případně rozsahu firmy. Další výhodou je i obsažená podpora ze strany dodavatele softwaru při implementaci nástroje. Součástí mnohdy bývá i procesní analýza firmy a optimalizace procesů s cílem maximálního využití nástroje PFEP. Z komerční nabídky lze zmínit například:

1. loopPFEP
2. Industry Star - Supply Performance System
3. Proplanner

[31, 23, 29]

2.3.2 Návrh nástroje

Pro návrh nástroje Plan for every part (PFEP) byla zvolena cesta tabulkových kalkulátorů. Konkrétně je nástroj vypracován v programu **Microsoft Excel 2016** z kancelářského balíku Microsoft Office. Toto s sebou nese mnohá pozitiva a znamená to například nulovou počáteční investici do softwarového balíku, protože je Microsoft Office standardním softwarovým vybavením pracovních stanic cílových osob ve firmě. Dále již zmíněná familiarita prostředí Excelu, která pomůže pracovníkům při seznamování se s nástrojem PFEP.

Obrázek 2.3: Uživatelské rozhraní Proplanner PFEP

Containers	Container Classes	Container Types	Clas	Cart Types	Groups	ID	Length	Width	Height	Dimension Unit	Max Weight	Empty Weight	Weight Unit	Group ID	Full Stack Quantity	Empty Stack Quantity	Color	
1						CART1	48.0000	48.0000	48.0000	Inches	3000.0000	300.0000	Pounds	SteelSec	1.0000		1.0000	Gray
2						EXD60606	6.0000	6.0000	7.0000	Inches	35.0000	2.0000	Pounds	Expendable	6.0000		6.0000	Brown
3						EXC363636	9.0000	9.0000	7.0000	Inches	250.0000	5.0000	Pounds	Expendable	2.0000		4.0000	Brown
4						EXP121206	12.0000	12.0000	6.0000	Inches	35.0000	2.0000	Pounds	Expendable	4.0000		5.0000	Brown
5						XL120705P	12.0000	7.0000	5.0000	Inches	35.0000	5.0000	Pounds	SteelTote	8.0000		8.0000	Green
6						XL-121507P	12.0000	15.0000	7.5000	Inches	35.0000	5.0000	Pounds	PlasticTote	6.0000		6.0000	Blue
7						XL-242214P	24.0000	22.0000	14.0000	Inches	35.0000	5.0000	Pounds	PlasticTote	3.0000		3.0000	Blue
8						XL-2962	48.0000	40.0000	34.0000	Inches	3000.0000	300.0000	Pounds	SteelBulk	2.0000		2.0000	Gray
9						XL-2965	63.0000	48.0000	34.0000	Inches	3000.0000	300.0000	Pounds	SteelBulk	2.0000		2.0000	Gray
10						XL-2968	95.0000	48.0000	34.0000	Inches	3000.0000	300.0000	Pounds	SteelBulk	2.0000		2.0000	Gray
11						XL-2970	32.0000	30.0000	25.0000	Inches	3000.0000	300.0000	Pounds	SteelBulk	2.0000		2.0000	Gray
12						XL-2973	32.0000	30.0000	32.0000	Inches	3000.0000	300.0000	Pounds	SteelBulk	2.0000		2.0000	Gray
13						XL-3225	32.0000	30.0000	25.0000	Inches	1200.0000	50.0000	Pounds	PlasticBulk	2.0000		2.0000	Gray
14						XL-481507P	48.0000	15.0000	7.5000	Inches	35.0000	5.0000	Pounds	PlasticTote	6.0000		6.0000	Blue
15						XL-4825	48.0000	40.0000	25.0000	Inches	1200.0000	50.0000	Pounds	PlasticBulk	2.0000		2.0000	Gray
16						XL-4850	48.0000	40.0000	50.0000	Inches	1200.0000	50.0000	Pounds	PlasticBulk	2.0000		2.0000	Gray
17																		
18																		
19																		
20																		
21																		
22																		

Zdroj: <https://www.proplanner.com/support/videos/pfep-overview-with-customer-perspectives>

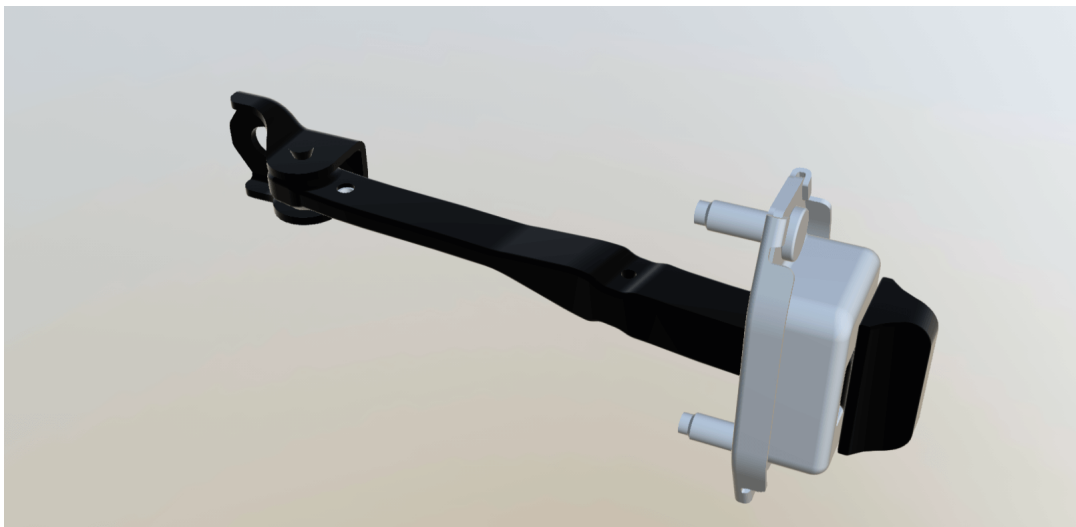
Struktura nástroje

Pro náš zjednodušený modelový scénář uvažujeme linku vyrábějící přední dvevní pant do automobilu, jehož vyobrazení je na obrázku 2.4 na straně 31. Finální montáž výrobku sestává ze sedmi dílů a tvoří tak sedm záznamů/řádků v nástroji PFEP. Nástroj je sestaven ze čtyř pracovníků listů, jejichž detail je možné zhlédnout v příloženém Excelu na CD nebo stáhnout na adrese <http://home.zcu.cz/~vanekr/DP/PFEP.zip> (Heslo pro .zip soubor je: zcufekpfep):

1. Úvodní list - Informace o provedených změnách nástroje PFEP
2. Data - Základní data a výpočty
3. RWT simulace - List pro výpočet a optimalizaci RWT
4. Obaly - Výpočet optimálního množství materiálu v L2 obale

Buňky v nástroji jsou rozděleny do tří kategorií. a to **tmavě šedé** - automaticky počítané buňky, **šedé** - automaticky počítané/uživatelsky vyplňované a bílé - uživatelsky vyplňované/importované.

Obrázek 2.4: Přední dveřní pant



Zdroj: Vlastní zpracování, 3D model z:
<https://grabcad.com/library/fr-door-check-link-1>

List - Úvodní list, Data

První list nástroje obsahuje informaci o změnách v nástroji, eviduje id změny, datum provedení změny, vykonavatele změny a schvalovatele změny. Následný list označený jako „Data“ obsahuje veškerá potřebná data konsolidována do jednoho listu z podnikového ERP a dalších podnikových systémů. Obsahuje také základní výpočty a logická pole. Celkový počet sedmadvadesáti sloupců je rozdělen do jedenácti kategorií. Sloupce jsou poté pojmenovány pro lepší orientaci ve vzorcích a jejich úpravu, správce názvů zachycuje obrázek 2.6 na straně 33. Prefixem definice názvu daného sloupce je název listu, který je následován podtržítkem a názvem hodnot.

Lookup - 2/97 Excelem počítaná pomocná pole zobrazující index řádku hodnoty v cizím listu, který je hledán pomocí materiálového čísla. Alokace samostatného sloupce pro nalezení indexu řádku v cizím listu výrazným způsobem urychlí vyhledání hodnot v případě velkého objemu dat.

Toto tvrzení bylo prověřeno pomocí testovacího Excelového souboru o třech instancích přístupu k hledání a nalezení hodnot. Každý list obsahoval dva referenční sloupce s 100 000 řádky uspořádaných dat, dále dvě dvojice sloupců pro vyhledávání. **Prvotní** test proběhl za užití funkce SVYHLEDAT, který byl dokončen za dvě minuty a osm vteřin. V **druhém** testu byla využita kombinace funkcí INDEX a POZVYHLEDAT bez separátního

Obrázek 2.5: Uživatelské prostředí nástroje

Lookup	Obecné	Logistika	Parametry dílu	Dodavatel	Transport	Výroba	Materiálový tok	Dispozice	Obaly	Náklady				
RWT simulace_lokup	Obaly_lookup	ID	Číslo dílu	Datum poslední změny	Název dílu	Disponent zk.	Hmotnost dílu 1 (kg)	Název dodavatele	Počet L1 obalů (indikátor)	Denní průměrný počet požadavků (ks)	Signál pro doplnění linky	Závod číslo	Typ L1 obalu	Náklady na dopravu jedné dodávky (EUR)
2	5	01056101714	02.05.2020	Arm_insert	NJA	230	VÍTKOVICE HEAV	0,89	1 400,00	WMS	3014	Europaleta		
3	6	01165623940	02.05.2020	Arm_injected	NJA	150	WITTE Nejdek, s.r.o.	0,78	1 400,00	WMS	3015	Europaleta		
4	7	01086752104	02.05.2020	Bracket	NJA	180	VÍTKOVICE HEAV	0,16	1 400,00	WMS	3016	Europaleta		
5	8	01194470930	02.05.2020	Pin_Bracket	NJA	8	KZL cz, s.r.o.	0,03	1 400,00	WMS	3017	Europaleta		
6	9	01192210641	02.05.2020	Cover	NJA	200	ALFE BRNO, s.r.o.	0,34	1 400,00	WMS	3018	Europaleta		
7	10	01049604346	02.05.2020	Case	NJA	100	ALFE BRNO, s.r.o.	0,30	1 400,00	WMS	3019	Europaleta		
8	11	01031732053	02.05.2020	Bolt	NJA	100	ALFE BRNO, s.r.o.	0,30	0,00	WMS	3020	Europaleta		

Zdroj: Vlastní zpracování

výpočtu indexu řádků. Výpočet byl dokončen za jednu minutu a šestačtyřicet sekund. **Poslední** test pracoval též s předešlou kombinací, ale s rozdílem výpočtu indexů pouze jednou a jeho následnou referencí do funkce INDEX. Finální výpočet byl proveden za osmatřicet vteřin.

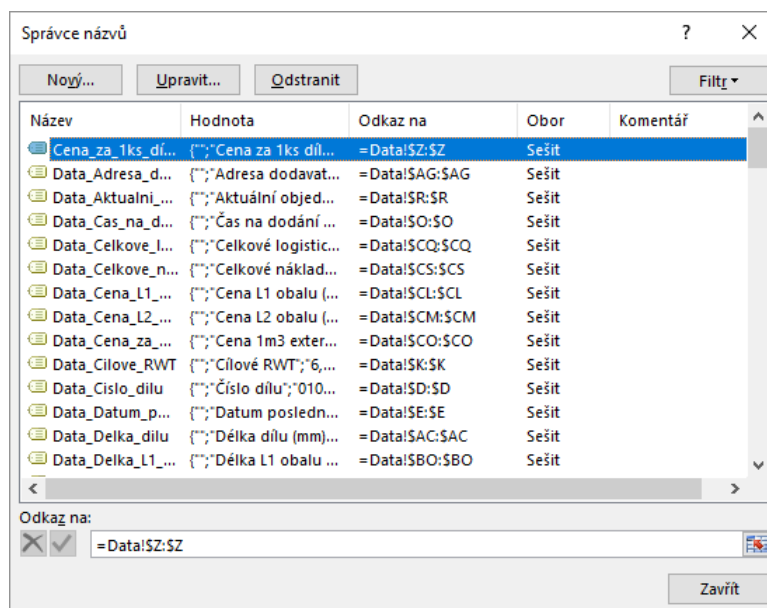
Obecné - 4/97 Kategorie obsahující základní informace o materiálu. Jedná se o ID záznamu, materiálové číslo z ERP, datum poslední změny informací o materiálu a název materiálu s obrazovým náhledem položky.

Logistika - 18/97 Obsahem této kategorie je název disponenta spravujícího daný materiál, dále znak ABC analýzy, současná hodnota RWT, cílová hodnota RWT převzata z listu simulace RWT. Následně pokračují informace ohledně počtu nastavených karenčních dnů, případné nastavení LSF a Minimum order quantity (MOQ).

Následuje aktuální objednávací množství L2 obalů⁵, optimální objednávací množství L2 obalů, které využívá pro svůj výpočet data ze simulace RWT a data o počtu dílů v L2 obalu. Dále optimální objednávací množství dílů, skladová zásoba, hodnota skladové zásoby vyjádřena

⁵Level two - krabice, přepravky atd.

Obrázek 2.6: Správce názvů



Zdroj: Vlastní zpracování

v eurech, stanovená frekvence objednávky ve dnech a optimální frekvence objednávky odvozená od cílového RWT ze simulace.

Parametry dílu - 7/97 Zde jsou obsaženy informace o hmotnosti dílu, jeho ceně, dále pak váha, zdali se nakupuje nebo vyrábí interně a rozměry dílu. Poslední položka je důležitá pro výpočet optimálního počtu materiálu v L2 obalu, který je popsán na straně 39.

Dodavatel - 9/97 Neméně důležité jsou informace o dodavateli daného materiálu. Zcela jistě zde patří název dodavatele, atomizované adresa⁶. Následně kontaktní údaje jako je e-mail a telefonní číslo. Pro jednoznačnou identifikaci evidujeme i identifikační číslo a interní hodnocení dodavatele.

Tato data mohou sloužit jako podklad pro různé Business Intelligence (BI) nástroje schopné tvořit analytické mapy a dashboardy. Můžeme zmínit například MS Power BI, Metabase či Tableau.

⁶Rozpadlá na dále nedělitelné části. Samostatně stát, město, ulice, PSČ atd.

Transport - 7/97 Kategorie transport obsahuje indikátor počtu L1 obalů⁷, který je roven poměru aktuálního objednáacího množství L2 a maximálního počtu L2 obalů na daném L1 obalu. Z tohoto vychází skutečný počet L1 obalů roven zaokrouhlenému číslu na horní nejbližší celé číslo,

Další položky jsou počet L2 obalů v dodaném L1 obale, počet dílů v dodaném L2 obale, ideální počet dílů v dodaném L2 obale (Vypočítaný z listu Obaly), dopravce a příjmový dok.

Výroba - 4/97 Informace o výrobě zahrnují denní průměrný počet požadavků, denní průměrnou hodnotu požadavků v eurech, maximální denní počet požadavků a počet zásobovaných montážních linek.

Materiálový tok - 8/97 Zde pracovníci naleznou informaci o signálu pro doplnění linky, způsobu doplňování na linku (Milk run, chaotic aj.), zásobování linky - obal, zásobování linky - kapacita obalu v kusech. Následuje výpočet obalů na směnu, výpočet obalů v běhu a atomizovaná lokace supermarketu.

Dispozice - 6/97 Podrobný rozpad lokace materiálu ve skladu závodu. Sestává z čísla závodu, názvu závodu, skladu, uličky ve skladu, sloupu uličky a patra.

Obaly - 23/97 Rozsáhlá skupina obsahující důležité informace o obalech. Nalezneme zde, typ L1 obalu, jeho délku, šířku, a výšku. Následuje hmotnost prázdného L1 obalu, nosnost L1 obalu, celková váha L1 + L2 a informace o překročení nosnosti L1 obalu.

Dále je zde maximální počet L2 obalů v/na L1 obalu, maximální počet pater na L1 obalu, výpočet pro determinaci počtu použitých vrstev L2 obalů. Poté kategorie uvádí informace o použitém typu L2 obalu, jeho délka, šířka a výška. Následuje hmotnost prázdného L2 obalu, vrstvy L2 obalu, které ovlivňují výpočet na listu „obaly“, celková výška L1 + L2 a hmotnost plného L2 obalu. Nosnost L2 obalu je zde též společně s překročením nosnosti L2 obalu.

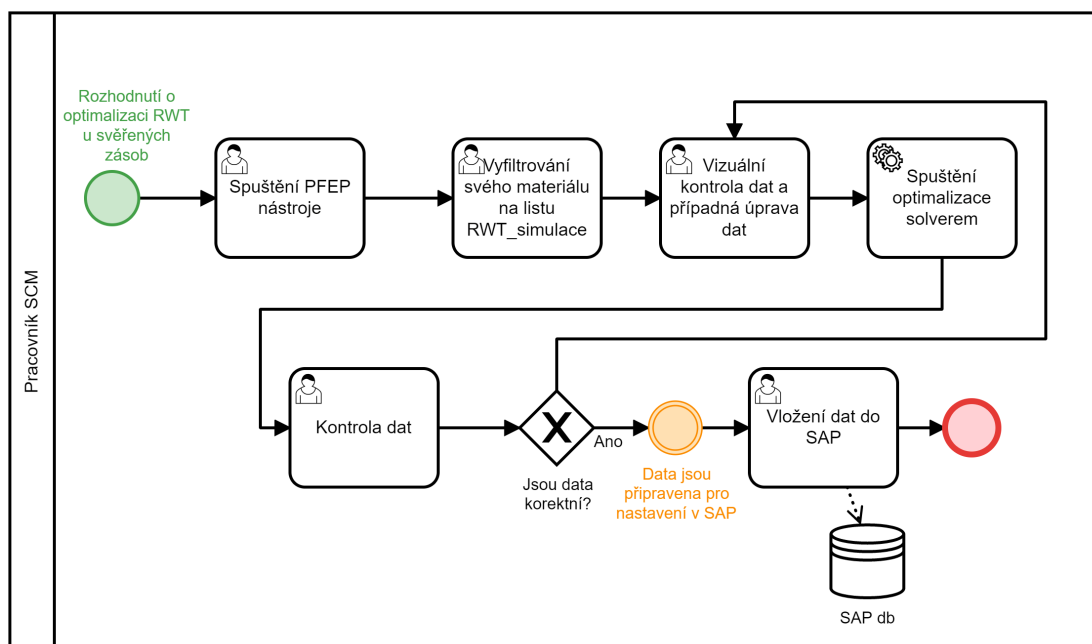
⁷Obaly pro přepravu, typicky palety, kontejnery atd.

Náklady - 9/97 Finální kategorie se zabývá náklady na materiál. Zastoupeny jsou zde náklady na dopravu jedné dodávky v eurech, cena L1 obalu, cena L2 obalu, náklady za jednu dodávku celkem. Poté cena 1m³ externího skladu, ostatní logistické náklady na 1ks L1 obalu, celkové logistické náklady, informace o použití externího skladu v m³ a celkové náklady na dodávku.

List - RWT simulace

Nyní se dostáváme k RWT simulaci a následné optimalizaci na třetím listu nástroje. Současný přístup pracovníků v řešení otázky optimalizace zásob a RWT je znázorněn na diagramu 1.10 na straně 22. Stávající neideální proces lze optimalizovat za pomoci PFEP a připraveného listu se simulací. Konsolidací dat z podnikových systémů do jednoho nástroje odpadá nutnost zdlouhavé přípravy dat ze SAP BW, jejich vkládání do Excel nástroje a následná úprava. Pracovník může rovnou přejít do listu simulace, vyfiltrovat si příslušný materiál dle dostupných kritérií, zkontrolovat parametry simulace a zahájit automatický výpočet. Návrh nového procesu je znázorněn na diagramu 2.7 na straně 35.

Obrázek 2.7: Návrh nového procesu simulace optimalizace RWT



Zdroj: Vlastní zpracování

Parametry simulace Opět zde nacházíme sloupec s vyhledávacím indexem, který je klíčový pro optimalizaci rychlosti nástroje zejména při značných objemech dat. Klíčové položky pro vyfiltrování a řazení materiálu jsou sloupce Disponent zk., Disponent, Znak analýzy ABC a případně RWT. Korektní kombinací filtrů a řazení získá disponent kýžená data pro optimalizaci zásob.

Manuální optimalizace za stávajícího stavu procesu ve WITTE Automotive (Witte) je zde nahrazena automatizací využívající řešitele úloh Lineárního programování(LP). Konkrétně byl v tomto případě použit externí řešitel OpenSolver⁸ nabízející vylepšené uživatelské rozhraní, větší výběr řešitelů a poskytuje vizuální (Náhled v příloze C) reprezentaci optimalizační úlohy v buňkách listu.

K úspěšnému použití řešitele je nutné nejprve zkonstruovat model matematického programování popisující řešenou situaci. Dle Plevnýho model matematického programování „vyjadřuje vztahy platné v reálném systému formou matematických výrazových prostředků, jako jsou: funkce, soustavy rovnice a nerovnic.“ [10, s. 16] Vstupy matematického modelu můžeme řadit na **řiditelné** a **neřiditelné**. Řiditelné vstupy figurují v modelu jako proměnné veličiny a jedná se například o:

- Počet vyráběných kusů výrobku
- Binární rozhodování ve formě Ano/Ne
- Velikost nákladu
- Velikost skladových zásob

Neřiditelné vstupy vystupují v modelu jako konstanty a patří mezi ně například:

- Kapacita výrobních zařízení
- Dny pro dodání materiálu
- Ceny nakupovaných surovin
- Spotřeba materiálu na jednotku produkce

Řešením matematického modelu je maximalizace či minimalizace dané účelové funkce za splnění všech omezujících podmínek. Cílem příkladové optimalizace je minimalizace účelové funkce, která představuje hodnotu stanovené cílové zásoby daného disponenta.

⁸<https://opensolver.org/>

Tabulka 2.2: Hodnoty modelu

	Cena za 1ks dílu	Čas dodání mat	Denní prů. hod. požadavků	MOQ
x1	1,53 €	6 dnů	2 142 €	8 000 ks
x2	2,12 €	7 dnů	2 968 €	9 800 ks
x3	3,40 €	7 dnů	4 760 €	9 000 ks
x4	1,10 €	5 dnů	1 540 €	12 500 ks
x5	4,20 €	6 dnů	5 880 €	7 200 ks
x6	5,10 €	9 dnů	7 140 €	10 500 ks
x7	0,70 €	12 dnů	1 960 €	25 000 ks

Zdroj: Vlastní zpracování

Proměnné modelu jsou jednotlivé plánované průměrné denní skladové zásoby⁹. V předchozím textu jsme byli obeznámeni se skutečností používání Just in time systému společností Witte. Budeme tedy v našem modelu omezení dodací dobou materiálu od dodavatelů do závodu Witte. Finální omezující podmínka je splnění/přiblížení se cílovému stanovenému globálnímu RWT nastavenému na hodnotu 5.

Zápis konkrétního matematického modelu z nástroje PFEP bude vypadat následovně:
minimalizuj

$$z = 1,53 * x1 + 2,12 * x2 + 3,40 * x3 + 1,10 * x4 + 4,20 * x5 + 5,10 * x6 + 0,70 * x7 \quad (2.1)$$

za podmínek:

$$\frac{1,53 * x1}{2 142} \geq 6 \quad (2.2)$$

$$\frac{2,12 * x2}{2 968} \geq 7 \quad (2.3)$$

$$\frac{3,40 * x3}{4 760} \geq 7 \quad (2.4)$$

$$\frac{1,10 * x4}{1 540} \geq 5 \quad (2.5)$$

$$\frac{4,20 * x5}{5 880} \geq 6 \quad (2.6)$$

$$\frac{5,10 * x6}{7 140} \geq 9 \quad (2.7)$$

⁹Podrobnosti modelu a simulace jsou obsaženy v příloženém Excel souboru.

KAPITOLA 2. POPIS NÁSTROJE PFEP

$$\frac{0,70 * x7}{1960} \geq 12 \quad (2.8)$$

$$\frac{1,53 * x1 + 2,12 * x2 + 3,40 * x3 + 1,10 * x4 + 4,20 * x5 + 5,10 * x6 + 0,70 * x7}{26390} \geq 5 \quad (2.9)$$

$$x1, x2, x3, x4, x5, x6, x7 \geq 0 \quad (2.10)$$

$$[x1, x2, x3, x4, x5, x6, x7 = int \quad (2.11)$$

Výpočtem pomocí OpenSolveru získáme $z = 19\,708$, $x1 = 8\,400$, $x2 = 9\,800$, $x3 = 9\,800$, $x4 = 7\,000$, $x5 = 8\,400$, $x6 = 12\,600$, $x7 = 33\,600$ a výsledné globální RWT = 7,49. Získali jsme tak optimální řešení za daných podmínek. Výsledné globální RWT však nedosahuje hodnot blízcích se stanovenému cíli. V grafickém zobrazení (2.8 na straně 38) může pracovník snadno nalézt položku s nejvyšším dílčím RWT, zkontrolovat omezující podmínky modelu, rozhodnout se, zdali existuje možnost úpravy omezujících podmínek a model opět spustit.

Obrázek 2.8: Výsledek optimalizace pomocí OpenSolver

Stanovené cílové RWT	Celková hodnota stanovené úspory (EUR)	Hodnota zbývající stanovené úspory ke splnění cíle (EUR)	Stanovená cílová zásoba (EUR)	Stanovená cílová úspora (EUR)	Stanovené cílové RWT	Celková hodnota stanovené úspory (EUR)	Hodnota zbývající stanovené úspory ke splnění cíle (EUR)			Omezující podmínky		
≤ 7,491777188	31 784,55	- 65 758,00	244 425,20	- 14 932,65	9,262038651	- 14 932,65	-112 475,20					
Denní průměrná hodnota požadavků (EUR)	RWT	Dílčí stanovené cílové RWT	Hodnota plánované úspory	Plánovaná denní průměrná hodnota skladové zásoby (EU)	Plánované denní průměrné skladové zásoby (ks)	RWT + poj. Zas. Delta	Dílčí stanovené cílové RWT + poj. Zas.	Hodnota plánované úspory + poj. Zas.	Plánovaná denní průměrná hodnota skladové zásoby + poj. Zas. (EUR)	Plánované denní průměrné skladové zásoby + poj. Zas. (ks)	Pojistná zásoba (ks)	Čas na dodání materiálu (dny)
2 142,00	7,80	6,00	3 854,07	12 852,00	8400	2,50	8,50	1 500,93	18 207,00	11 900,00	3 500,00	6
2 968,00	8,11	7,00	3 305,08	20 776,00	9800	2,86	9,86	5 174,92	29 256,00	13 800,00	4 000,00	7
4 760,00	9,76	7,00	13 158,00	33 320,00	9800	1,67	8,67	5 225,80	41 252,20	12 133,00	2 333,00	7
1 540,00	9,92	5,00	7 579,00	7 700,00	7000	2,14	7,14	4 279,00	11 000,00	10 000,00	3 000,00	5
5 880,00	9,43	6,00	20 147,40	35 280,00	8400	1,79	7,79	9 647,40	45 780,00	10 900,00	2 500,00	6
7 140,00	7,82	9,00	8 445,60	64 260,00	12600	1,07	10,07	16 095,60	71 910,00	14 100,00	1 500,00	9
1 960,00	8,01	12,00	7 813,40	23 520,00	33600	1,79	13,79	11 313,40	27 020,00	38 600,00	5 000,00	12

Zdroj: Vlastní zpracování

[10]

Rozšířením modelu o Minimum order quantity (MOQ) získáme přídavek těchto omezujících podmínek:

$$x_1 \geq 8\,000 \quad (2.12)$$

$$x_2 \geq 9\,800 \quad (2.13)$$

$$x_3 \geq 9\,000 \quad (2.14)$$

$$x_4 \geq 12\,500 \quad (2.15)$$

$$x_5 \geq 7\,200 \quad (2.16)$$

$$x_6 \geq 10\,500 \quad (2.17)$$

$$x_7 \geq 25\,000 \quad (2.18)$$

Provedeme-li výpočet za těchto podmínek, získáme $z = 20\,3758$, $x_1 = 8\,400$, $x_2 = 9\,800$, $x_3 = 9\,800$, $x_4 = 12\,500$, $x_5 = 8\,400$, $x_6 = 12\,600$, $x_7 = 33\,600$ a výsledné globální RWT = 7,721. Příslušný pracovník může změnou omezujících podmínek a použitím funkce scénářů v Excelu provést citlivostní analýzu a vyhodnotit potenciální kroky k optimalizaci zásob svěřeného materiálu. Rozhodne se na základě dat, zdali má zahájit proces vyjednávání s dodavatelem ohledně možnosti snížit dodací dobu či snížit/navýšit MOQ

List - Obaly

Poslední list obsahuje výpočet optimálního počtu kusů materiálu v L2 obale. List obsahuje opět pomocný sloupec pro nalezení indexu řádku z listu „Data“ z důvodu optimalizace rychlosti běhu výpočtů. Následuje sloupec s informací o čísle dílu, který je použit pro výpočet prvního sloupce, počet vrstev v L2 obale a sloupec s hodnotou tolerance stěn. Tato hodnota nám udává rozdíl mezi vnějšími rozměry a vnitřními rozměry L2 obalu. Pro zjednodušení je zde brána hodnota třech milimetrů na jednotlivou vrstvu L2 obalu, která poté vstupuje do vzorců pomocných sloupců a ponižuje zde rozměry L2 obalu.

Následují tři sady sloupců pro pomocné výpočty:

1. Délka L2/ Délka dílu, Šířka L2 / Délka dílu, Výška L2/ Délka dílu
2. Délka L2 / Šířka dílu, Šířka L2 / Šířka dílu, Výška L2 / Šířka dílu
3. Délka L2 / Výška dílu, Šířka L2 / Výška dílu, Výška L2/ Výška dílu

Výpočet optimálního množství kusů je proveden pomocí Visual Basic for Applications (VBA) skriptu, který porovnává hodnoty definovaných sad sloupců a vrací jejich maxima při zachování podmínek orientace materiálu v prostoru. Skript vypočítá ideální počet kusů materiálu v L2 obale z různých kombinací orientace materiálu v obale. Pro zjednodušení předpokládáme maximální naměřené hodnoty délky, výšky a šířky materiálu jako délky entity kvádru, která vstupuje do výpočtu.

```

1 Sub Obaly()
2 Dim rg1 As Range
3 Set rg1 = ActiveSheet.Range("F:N").CurrentRegion
4
5 Dim rg2 As Range
6 Set rg2 = ActiveSheet.Range("P:S").CurrentRegion
7
8 Dim i As Long
9 Dim arr(9) As Integer
10 Dim arr1(3) As Integer
11 Dim arr2(3) As Integer
12 Dim arr3(3) As Integer

```

V této části kódu definujeme proceduru „Obaly“, kterou poté budeme v listu Excelu volat pomocí spárování s tlačítkem „Vypočítat ideální počet dílů v obalu“. Dále definujeme dvě oblasti dat, a to „F:N“ ukládané do proměnné „rg1“ a „P:S“ ukládané do proměnné „rg2“. Dále definujeme pomocnou proměnnou „i“ a zvolíme datový typ „Long“¹⁰. Následně definujeme čtyři pole, „arr“ s délkou devíti pozic, a „arr1, arr2, arr3“ s délkou tří pozic.

```

13 For i = 2 To rg1.Rows.Count
14
15     'Definice globálního pole
16     arr(0) = rg1.Cells(i, 1).Value2
17     arr(1) = rg1.Cells(i, 2).Value2
18     arr(2) = rg1.Cells(i, 3).Value2
19     arr(3) = rg1.Cells(i, 4).Value2

```

¹⁰Datový typ „Long“ slouží pro ukládání velkých celých čísel, která přesahují limit datového typu „Integer“


```
20 arr(4) = rg1.Cells(i, 5).Value2
21 arr(5) = rg1.Cells(i, 6).Value2
22 arr(6) = rg1.Cells(i, 7).Value2
23 arr(7) = rg1.Cells(i, 8).Value2
24 arr(8) = rg1.Cells(i, 9).Value2
25
26 'Definice pole1
27 arr1(0) = rg1.Cells(i, 1).Value2
28 arr1(1) = rg1.Cells(i, 2).Value2
29 arr1(2) = rg1.Cells(i, 3).Value2
30 'Definice pole2
31 arr2(0) = rg1.Cells(i, 4).Value2
32 arr2(1) = rg1.Cells(i, 5).Value2
33 arr2(2) = rg1.Cells(i, 6).Value2
34 'Definice pole3
35 arr3(0) = rg1.Cells(i, 7).Value2
36 arr3(1) = rg1.Cells(i, 8).Value2
37 arr3(2) = rg1.Cells(i, 9).Value2
```

Zde inicializujeme „For loop“ (Smyčku), který začíná hodnotou 2 z důvodu přeskočení prvního řádku s hlavičkou. Hodnota proměnné „i“ je navyšována do konečného počtu řádků námi stanoveného pole „rg1“. Kód je zaměřen primárně na funkčnost a proto zde definujeme a plníme pole v celkovém počtu čtyři. Bohužel VBA nedisponuje elegantním řešením extrakce intervalu z pole jako například Python a jiné programovací jazyky. Nelze tedy snadným způsobem extrahovat intervaly hodnot pouze z jednoho pole, zde tedy „arr“. Definice tří polí není zcela optimální řešení z hlediska programování jako takového, ale je to řešení funkční.

```
38 'Hledání maxima globálního pole
39 max_val = WorksheetFunction.Max(arr)
40 max_val_index = Application.Match(max_val, arr, False)
41 switch_arr1 = arr1
42 switch_arr2 = arr1
43 switch_arr_str1 = 1
44 switch_arr_str2 = 1
```

Tato část skriptu definuje proměnnou „max_val“, která obsahuje hodnotu globálního maxima z aktuálního pole „arr“. Proměnná „max_val_index“ poté obsahuje index hodnoty globálního maxima, který nám poslouží k určení výběru polí pro hledání lokálních maxim. Následně definujeme pomocné proměnné.

```
45 'První interval má globální max
46 If max_val_index <= 3 Then
47     local_max1 = WorksheetFunction.Max(arr2)
48     local_max2 = WorksheetFunction.Max(arr3)
```

```

49     switch_arr1 = arr2
50     switch_arr2 = arr3
51     switch_arr_str1 = 2
52     switch_arr_str2 = 3

```

Globální maximum se nachází v prvním intervalu v případě, že „max_val_index“ nabývá hodnot 0, 1 nebo 3. Dále definujeme proměnné pro lokální maxima zbylých polí „local_max1, local_max2“, které hledají maxima v polích „arr2, arr3“. Aktualizujeme pomocná pole.

```

53     If Application.Match(local_max1, arr2, False) = 1 And max_val_index = 1 Then
54         local_max1 = WorksheetFunction.Large(arr2, 2)
55     ElseIf Application.Match(local_max1, arr2, False) = 2 And max_val_index = 2 Then
56         local_max1 = WorksheetFunction.Large(arr2, 2)
57     ElseIf Application.Match(local_max1, arr2, False) = 3 And max_val_index = 3 Then
58         local_max1 = WorksheetFunction.Large(arr2, 2)
59     End If
60
61     If Application.Match(local_max2, arr3, False) = 1 And max_val_index = 1 Then
62         local_max2 = WorksheetFunction.Large(arr3, 2)
63     ElseIf Application.Match(local_max2, arr3, False) = 2 And max_val_index = 2 Then
64         local_max2 = WorksheetFunction.Large(arr3, 2)
65     ElseIf Application.Match(local_max2, arr3, False) = 3 And max_val_index = 3 Then
66         local_max2 = WorksheetFunction.Large(arr3, 2)
67     End If
68 End If

```

V této části kódu kontrolujeme poziční hodnotu lokálních maxim s poziční hodnotou (indexem) globálního maxima. V případě shody je hodnota „local_max1/2“ aktualizována na druhou nejvyšší hodnotu v daném poli.

```

69     'Druhý interval má globální max
70     If max_val_index > 3 And max_val_index <= 6 Then
71         local_max1 = WorksheetFunction.Max(arr1)
72         local_max2 = WorksheetFunction.Max(arr3)
73         switch_arr1 = arr1
74         switch_arr2 = arr3
75         switch_arr_str1 = 1
76         switch_arr_str2 = 3
77
78     If Application.Match(local_max1, arr1, False) = 1 And max_val_index = 4 Then
79         local_max1 = WorksheetFunction.Large(arr1, 2)
80     ElseIf Application.Match(local_max1, arr1, False) = 2 And max_val_index = 5 Then
81         local_max1 = WorksheetFunction.Large(arr1, 2)
82     ElseIf Application.Match(local_max1, arr1, False) = 3 And max_val_index = 6 Then
83         local_max1 = WorksheetFunction.Large(arr1, 2)
84     End If
85
86     If Application.Match(local_max2, arr3, False) = 1 And max_val_index = 4 Then

```

```

87         local_max2 = WorksheetFunction.Large(arr3, 2)
88     ElseIf Application.Match(local_max2, arr3, False) = 2 And max_val_index = 5 Then
89         local_max2 = WorksheetFunction.Large(arr3, 2)
90     ElseIf Application.Match(local_max2, arr3, False) = 3 And max_val_index = 6 Then
91         local_max2 = WorksheetFunction.Large(arr3, 2)
92     End If
93 End If
94
95 'Třetí interval má globální max
96 If max_val_index > 6 Then
97     local_max1 = WorksheetFunction.Max(arr1)
98     local_max2 = WorksheetFunction.Max(arr2)
99     switch_arr1 = arr1
100    switch_arr2 = arr2
101    switch_arr_str1 = 1
102    switch_arr_str2 = 2
103
104    If Application.Match(local_max1, arr1, False) = 1 And max_val_index = 7 Then
105        local_max1 = WorksheetFunction.Large(arr1, 2)
106    ElseIf Application.Match(local_max1, arr1, False) = 2 And max_val_index = 8 Then
107        local_max1 = WorksheetFunction.Large(arr1, 2)
108    ElseIf Application.Match(local_max1, arr1, False) = 3 And max_val_index = 9 Then
109        local_max1 = WorksheetFunction.Large(arr1, 2)
110    End If
111
112    If Application.Match(local_max2, arr2, False) = 1 And max_val_index = 7 Then
113        local_max2 = WorksheetFunction.Large(arr2, 2)
114    ElseIf Application.Match(local_max2, arr2, False) = 2 And max_val_index = 8 Then
115        local_max2 = WorksheetFunction.Large(arr2, 2)
116    ElseIf Application.Match(local_max2, arr2, False) = 3 And max_val_index = 9 Then
117        local_max2 = WorksheetFunction.Large(arr2, 2)
118    End If
119
120 End If

```

Zde probíhá vyhledávání a kontrola pozice lokálních maxim v případě lokace globálního maxima v druhé a třetí sadě sloupců.

```

121     If (Application.Match(local_max1, switch_arr1, False) = Application.Match(local_max2,
122         ↪ switch_arr2, False)) And local_max1 < local_max2 Then
123         local_max1 = WorksheetFunction.Large(switch_arr1, 3)
124     ElseIf (Application.Match(local_max1, switch_arr1, False) = Application.Match(local_max2,
125         ↪ switch_arr2, False)) And local_max1 > local_max2 Then
126         local_max2 = WorksheetFunction.Large(switch_arr2, 3)
127
128 End If

```

Kontrola shodnosti pozičních indexů lokálních proměnných. V případě shodných hodnot je zvolena třetí největší (Ze tří celkových elementů ta nejmenší) hodnota pro takové lokální maximum, které je v porovnání nižší.

```

127     rg2.Cells(i, 1).Value2 = WorksheetFunction.Max(arr)
128     rg2.Cells(i, 2).Value2 = local_max1
129     rg2.Cells(i, 3).Value2 = local_max2
130
131 Next i
132 End Sub

```

Finální zápis hodnot do buněk, který je následován příkazem pro přesun k další hodnotě „i“ a celý „For loop“ (smyčka) se opakuje do finální konečné hodnoty „i“.

Možnost využití OpenSolver/Řešitele Zkusme nyní porovnat předchozí řešení s možným řešením za pomoci OpenSolveru, který jsme použili při řešení simulace RWT na straně 35. Zkonstruovaný model a jeho výpočet nebude zcela shodný s výpočtem skriptu z důvodu nemožnosti popsat tyto operace matematickým modelem. Kontrolu pozičního indexu globálního, lokálních maxim a výpočet účelové funkce zajistíme s použitím pomocných sloupců. Tři sloupce s pozičními indexy, které budou proměnné modelu (O:Q) a pomocný sloupec obsahující pro-násobení těchto tří sloupců (R). Hodnota tohoto sloupce musí být rovna šesti, protože se pozice maxim nesmí opakovat (Tzn. $3 * 2 * 1 = 6$). Sloupec dílčích hodnot řádků obsahuje vynásobené funkce „INDEX“, které přijímají za argumenty příslušné oblasti a indexy z pomocných sloupců (S). Cílem modelu je maximalizovat sumu těchto dílčích hodnot.

Obrázek 2.9: NP model optimalizace počtu kusů materiálu v L2 obalu

	O	P	Q	R	S	T
	Look1	Look2	Look3		Sum	Max Sum
	1	1	1	1	506	37781
	1	1	1	1	360	
	1	1	1	1	1144	
3≥,1≤	1	1	1	6=	17424	
	1	1	1	1	1332	
	1	1	1	1	680	
	1	1	1	1	16335	

Zdroj: Vlastní zpracování

Při spuštění Solveru se bohužel dočkáme dlouhé prodlevy pro výpočet. Pro názornost byla provedena série porovnávacích testů, ve kterých se postupně přidávaly řádky pro výpočet.

Tabulka 2.3: Porovnání exekuce VBA skriptu a Řešitele

Počet řádků	Řešitel	VBA
3	0,672 s	0,3 s
4	52,2 s	0,3 s
5	180,5 s	0,3 s
6	638,6 s	0,3s
7	5 295,4 s	0,3 s
500	-	0,89 s
1 000	-	1,8 s
10 000	-	18,41 s
100 000	-	181 s

Zdroj: Vlastní zpracování

Výsledky jsou zachyceny v tabulce 2.3 na straně 45. Z hodnot je patrná nevhodnost použití Řešitele na tento typ problému.

3. Postup implementace PFEP

3.1 Kroky implementace

V kapitole 2 na straně 23 byla probrána historie Toyota production system (TPS), podmínky vzniku, představena základní myšlenka tohoto systému a byl popsán návrh PFEP nástroje vypracovaného v Excelu. Obecně lze říct, že jeden z hlavních cílů TPS je redukce plýtvání (**Muda**), zamezení výpadkům výroby (**Mura**) a to vše bez přetížení (**Muri**). K dosažení kýženého cíle TPS identifikuje několik nástrojů jako je Value stream mapping (VSM), Kaizen, 5s, SMED, Kanban, ale i PFEP. Potenciál TPS je nedosažen bez přesné, relevantní a aktuální informace o materiálovém toku firmy.

Jim Womack o klíčové roli PFEP hovoří následovně: „*What’s needed instead is system kaizen in which the material-handling system for an entire facility, supplying every value stream, is redesigned to create a bulletproof delivery process that is utterly precise and stable. Such a system must include a plan for every part (PFEP) that documents all relevant information about each part number in the facility, including its storage location and points of use. It must also include precisely designed supermarkets, both for purchased parts and for work-in-process, that assign each part number a single storage location and minimum and maximum inventory quantities.*“ [14, s. 44]

[2, 3, 15]

3.1.1 Předpoklady

Systém TPS je možno přirovnat k voru, pomyslná suma jeho částí je pro fungování systému hodnotnější jako jeho jednotlivé části. PFEP je v našem přirovnání pomyslné lano, které upevňuje správné části na správném místě a umožňuje voru plout jako celek. Bohužel je pomyslné lano často při implementacích TPS a štíhlé výroby opomíjeno a firmy tak nedosahují požadovaných výsledků.

Při implementaci PFEP musíme pamatovat na předpoklady technické a uživatelské.

Uživatelské předpoklady

Práce s lidmi je při implementaci klíčová a odvíjí se od ní výsledek snažení implementačního týmu. Můžeme se setkat s následujícími postoji pracovníků:

1. Optimista - Lidé schopni vnímat vizi a přínosy PFEP pro firmu. Většinou se jedná o osoby do jisté míry již znalé dané problematiky.
2. Neutrální - Fráze „nejsem si jist, nevím, uvidíme“ patří do slovníku pracovníků patřící do této kategorie. Aktivně nesabotují implementaci, ale vyčkávají na výsledek a své implementační činnosti vykonávají s neutrálním přístupem
3. Pesimista - Velice negativní a vokální lidé, kteří implementaci nevěří a šíří firmou pomluvy. S tímto je nutné pracovat a zamezit šíření negativních zpráv, protože mohou poškodit samotnou implementaci.

Technické předpoklady

Druhou částí jsou technické předpoklady k provedení implementace PFEP. Jedná se o:

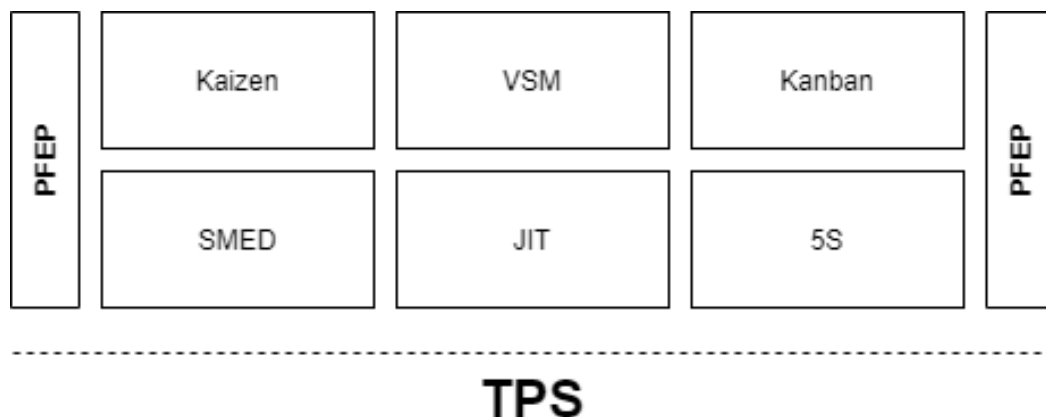
1. Výpočetní techniku
2. Ostatní

Firma musí disponovat základní počítačovou výbavou pro své pracovníky a v našem případě disponovat i kancelářským balíkem Microsoft Office, konkrétně Excel. Nástroj PFEP vytvořený v Excelu musí komunikovat s externími systémy, které slouží jako zdroj aktuálních dat. Pomyslný vor však nevytvoříme pouze s lanem (PFEP). Příslušné pomyslné dřevěné klády zde figurují jako druhá podmínka plavby na voru. [2, 3, 15]

3.1.2 Kroky tvorby nástroje

Vytvořený nástroj je popsán blíže v kapitole 2 na straně 23. Tvorba nástroje je specifická pro každou firmu, protože se liší požadavky pracovníků na funkcionalitu nástroje. Nicméně lze cítit obecný postup, který je aplikovatelný na všechny implementace.

Obrázek 3.1: TPS + PFEP



Zdroj: Vlastní zpracování

Identifikace vstupních dat

Unikátní požadavky každé firmy znamenají jedinečnou podobu nástroje v každé firmě a nelze tedy použít obecnou šablonu. Výslednou podobu bude diktovat základna dat a požadavky na výstupní hodnoty a případné simulace. Minimální datová základna bude tvořena již zaběhlými procesy a jejich vstupními daty.

Pamatujme však i na budoucí vývoj a možný cílový stav firmy za rok, dva i pět let. Fáze identifikace vstupních dat bude jistě benefitovat z kreativních brainstormingových schůzek generujících mnoho nápadů. Vhodná poučka pro přemýšlení nad daty je jejich atomicita, neboli rozdělení na nejmenší nedělitelné jednotky. Příkladem může být rozdělení rozměrů obaly do tří sloupců místo jednoho. Náročná, ale velice důležitá fáze bude zakončena sepsáním požadavků a přání pracovníků a managementu. [2, 3, 15]

Tvorba nástroje

Po fázi definice požadavků na výstupy PFEP nastává fáze tvorby nebo nákupu již existujícího softwarového řešení. Prvotní rozbor požadavků odpoví na základní otázku volby zmiňované cesty v kapitole 2.3.1 na straně 65.

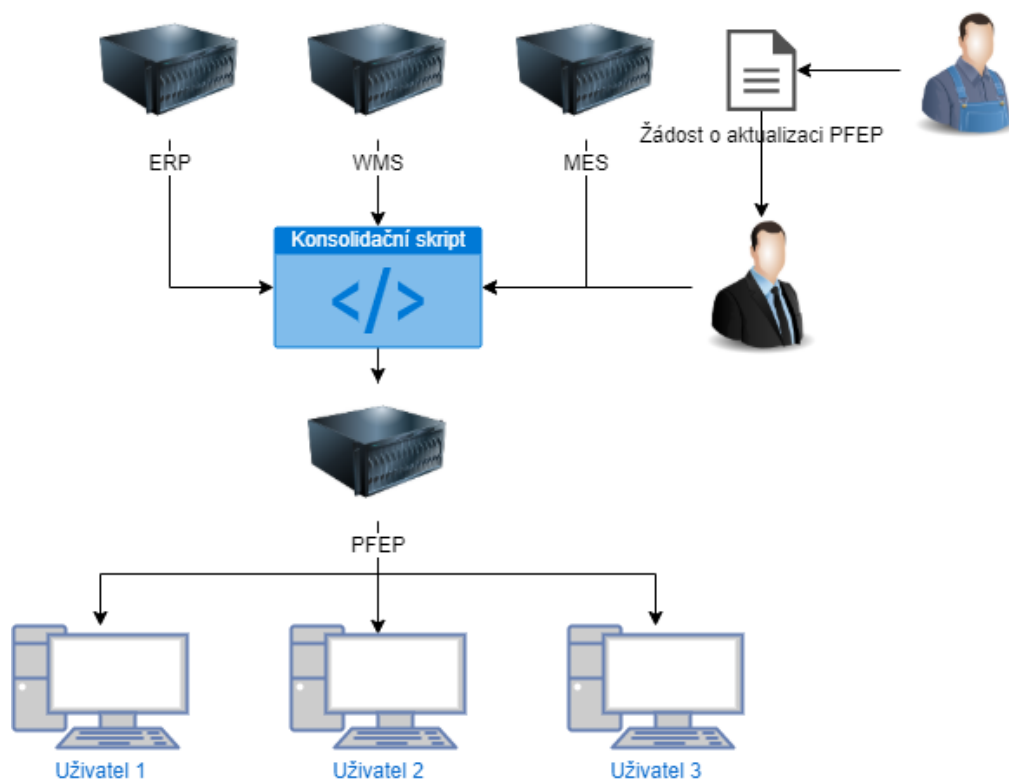
Nákup existujícího řešení může nabývat smyslu při nevyhovujícím složení schopného personálu, který by takto náročný úkol zvládl dokončit. Externí firma specializující se na vývoj PFEP nástrojů čerpá ze svých předešlých zkušeností a může nabídnout i cenný

externí pohled na procesy firmy a nalézt případná úzká místa. Nevýhodou nákupu specializovaného softwaru je iminentní přizpůsobování softwaru pro potřeby firmy, které může být problematické a finančně nevýhodné.

Tvořit nástroj interně je možno v případě menší komplexnosti požadovaných výstupů a dostatečného počtu pracovníků, které lze alokovat na tvorbu PFEP. Většina firem zvolí při interním vývoji některý z dostupných tabulkových kalkulátorů jako základ svého nástroje. Flexibilita a uživatelská přívětivost tabulkových kalkulátorů pomůže v rapidním vývoji. Nicméně i při užití Excelu je vhodné myslet na případnou budoucí migraci do prostředí databází.

Pomyslnou tepnou PFEP je zdroj dat firmy. Ve většině případů se bude jednat o různé Enterprise resource planning (ERP), Manufacturing Execution Systems (MES) a Warehouse management system (WMS). Tok dat je zobrazen na diagramu 3.2 na straně 49. Po vytvoření a nastavení PFEP workflow je vhodné vypracovat návodky a uspořádat školení pro uživatele.

Obrázek 3.2: Tok dat pro potřeby PFEP



Zdroj: Vlastní zpracování

3.1.3 Management PFEP

Životaschopnost podnikového softwaru stojí a padá na uživatelích a nastavených procesech. Tvorba nástroje a jeho poskytnutí uživatelům nepřinese benefity samo o sobě. Velkou pozornost musíme věnovat i promyšlení pracovního workflow s PFEP a ustanovení odpovědné osoby spravující PFEP nástroj a metodiku s ním spjatou. Správné fungování nástroje se odvíjí od kvality vstupních dat a od dodržování stanovených postupů.

Vlastník PFEP

Conrad hovoří o správě nástroje takto: „*The PFEP is a living document and the DNA of your value stream(s). Therefore, maintenance and management of the PFEP is critical to ensure that there is one version of the truth.*“ [2, s. 17] Udělení odpovědnosti za PFEP jedinému manažerovi bude mít za následek ať už osobní satisfakci pracovníka pro takto důležitou roli, taktéž firma získá aktivně řízený a spravovaný cenný nástroj. Pocit odpovědnosti a chtíč uspět požene pracovníka ke svědomité a dobře odvedené práci. Hlavní činností PFEP manažera je také zajištění integrity a aktuálnosti dat. Obsah PFEP by měl být pro čtení snadno dostupný všem pracovníkům ve firmě, výsadní právo editace, změn a aktualizace PFEP má pouze jediný člověk.

Při vybírání odpovědné osoby pamatujme na skutečnost, že lokální optimalizace nemusí nutně implikovat optimalizaci globální. Zvolení technika za PFEP manažera může vést ke zlepšení procesu v oblasti konstrukce, ale taková změna může mít za následek noční můru pro obalové specialisty, výrobní dělníky atd. Manažer PFEP musí cítit i odpovědnost za celý hodnotový řetězec a mít celopodnikový nadhled. Zdlouhavé a náročné přípravy nástroje PFEP nesou ovoce v rozhodovacím procesu založeném na datech, nikoliv na pocitu. Důvody pro držení zásob jsou jednoznačné a vysvětlitelné. Takový je cílový stav po implementaci PFEP.

4. Ekonomické zhodnocení zavedení

PFEP

V první kapitole této práce jsme se seznámili s klíčovými otázkami logistických nákladů. Z interních statistik společnosti WITTE Automotive (Witte) vyplývá podíl zásob na aktivech společnosti v rozmezí 16% a 22% za roky 2006 až 2018, viz graf 1.5 na straně 14. I malé snížení zásob může mít značný vliv na hospodaření společnosti. Společně s náklady na pořízení materiálu zásoby přináší například i náklady na skladovací prostory, obalový materiál, manipulaci, dopravu a jiné. Všechny tyto a jiné komponenty tvoří celkové logistické náklady, které jsou součástí cenových kalkulací a ovlivňují tak i cenovou konkurenceschopnost firmy.

Jak již bylo řečeno v kapitole 1, Witte pracuje v rámci řízení a optimalizace zásob s několika nástroji. Jsou to například pojistné zásoby, karenční dny a jiné. Hlavním ukazatelem a jedním z KPI celého Witte group je Reichweite (RWT), jehož výpočet je zachycen v rovnici 1.6 na straně 18. V rámci ekonomického zhodnocení se zaměříme právě na tento ukazatel a porovnáme **scénář 0 - PFEP nezaveden** se **scénářem 1 - PFEP zaveden**. K provedení zhodnocení byla vytvořena jednoduchá simulace ve skriptovacím jazyce Python, která supluje velice obtížné porovnání scénářů na reálných datech. Předložený nástroj v předchozí kapitole pohlíží na zásoby z několika dimenzí jako jsou například optimální počet kusů materiálu v L2 obale, váha, kategorie a jiné. Navržená simulace pracuje s hodnotami, které přímo ovlivňují RWT jako jsou dny na dodání materiálu, průměrné denní požadavky na materiál atd. Pro zjednodušení zde neuvažujeme omezující podmínku Minimum order quantity (MOQ). Nicméně i za těchto předpokladů poskytuje simulace relevantní data. Simulaci lze spustit pomocí přiloženého kódu v příloze B nebo na adrese (Při prvotním spuštění je nutné vyčkat na instalaci knihoven, graf je uložen v souborech, ikona souboru) <http://home.zcu.cz/~vanekr/DP/simulace.html>

Následující sekce popisuje interní logiku simulace a přibližuje její fungování. Spuštění a výsledky simulace se nachází v části 4.0.2 na straně 57

4.0.1 Rozbor skriptu simulace

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Sun Apr 26 18:24:44 2020
4
5 @author: richard.vanek
6 """
7 import time, random, csv, statistics, matplotlib.pyplot as plt, pyinputplus as pyip
8 from pulp import *
9 start_time = time.time()
10 print('Podpůrná simulace pro diplomovou práci "Studie logistického nástroje"\n Autor: Richard
    ↪ Vaněk\n')
11 automat = pyip.inputChoice(['1', '0'],prompt="Zadejte hodnotu 1 pro automatickou simulaci, hodnotu 0
    ↪ pro simulaci s uživatelským vstupem")
12 if automat == "1":
13     dolni_hrac = pyip.inputNum("Zadejte Dolní hranice variability automatického hráče. Stiskněte
    ↪ ENTER pro výchozí hodnotu: 0.2", min=0.01, default=0.2, lessThan=1, limit=1)
14     horni_hrac = pyip.inputNum("Zadejte Horní hranice variability automatického hráče. Stiskněte
    ↪ ENTER pro výchozí hodnotu: 0.8", min=0.01, default=0.8, lessThan=1, limit=1)
15 else:
16     pass

```

První řádky skriptu simulace slouží pro import důležitých knihoven pro běh simulace. Knihovna **random** slouží pro generování pseudo náhodných čísel Mersenne Twister generátorem. Knihovna **random** není vhodná pro kryptografické účely, ale možnost využít tzv. „seeds“ k momentálnímu „zmražení“ stavu generátoru je pro účely naší simulace ideální, protože umožňuje replikovat výsledky běhu simulace. Dále je importována knihovna pro statistické výpočty „**statistics**“.

Knihovna „**csv**“ má na starosti čtení a zápis souborů ve formátu csv. K vytvoření výsledného grafu je použita populární knihovna „**matplotlib**“, která nabízí širokou škálu možností vizualizace dat od jednoduchých sloupcových grafů až po 3D grafy, spektrogramy a jiné. Knihovna „**pyinputplus**“ poskytuje nadstavbu vestavěné funkci „**input()**“ v podobě ověření vstupních dat (Zdali uživatel zadal celé číslo atd.) Pro výpočet běhu simulace je použita knihovna „**time**“. Posledním importem je knihovna „**pulp**“, která zajišťuje funkce pro definici a výpočet lineárních matematických optimalizačních úloh. Po importu následuje definice parametrů simulace. Proměnná „**automat**“ požaduje po uživateli údaj o automatickém běhu simulace (Hodnota 1) nebo o uživatelském běhu simulace (Hodnota 0). V případě automatického běhu simulace je nutné zadat parametry variability počítačového hráče, proměnné „**dolni_hrac**“ a „**horni_hrac**“.

KAPITOLA 4. EKONOMICKÉ ZHODNOCENÍ ZAVEDENÍ PFEP

```
17 variabilita_pozadavku = pyip.inputNum("Zadejte variabilitu poptávky materiálu. Stiskněte ENTER pro  
↪ výchozí hodnotu: 0.1", min=0.01, default=0.1, lessThan=1, limit=1)  
18 pocet_obdobi = pyip.inputInt("Zadejte počet období. Stiskněte ENTER pro výchozí hodnotu: 12", min=1,  
↪ default=12, limit=1)  
19 variabilita_casu = pyip.inputNum("Zadejte variabilitu časů. Stiskněte ENTER pro výchozí hodnotu:  
↪ 0.4", min=0.01, default=0.4, lessThan=1, limit=1)  
20 skore_pocitace_var = []  
21 skore_pocitace = []  
22 skore_hrace = []  
23 sum_hodnoty_zasob = []  
24 sum_potreby = []  
25 sum_pokuta = []  
26 csv_export = []  
27 RWT_list = []  
28 LP_promenne = []  
29  
30 # Uživatelsky plánované denní průměrné skladové zásoby  
31 prum_sklad_zas = {"Arm_insert" : 8500, "Arm_injected" : 10500, "Bracket" : 10800, "Pin_Bracket" :  
↪ 12500, "Cover" : 7800, "Case" : 12600, "Bolt" : 35000}
```

Dále žádáme uživatele o informaci pro určení variability poptávky po materiálu, počtu období a variability časů šoku na dodací dobu materiálu. Hodnoty neoptimálního hráče jsou odvíjeny od optimálního výpočtu užitkové funkce LP modelu, které jsou násobeny zvoleným modifikátorem. V druhé polovině skriptu definujeme pomocné listy.

```
32 for i in range(pocet_obdobi + 1):  
33     # Definice random seedu  
34     random.seed(a=random.random(), version=2)  
35     # Definice hodnot pro LP úlohu  
36     goal_RWT = random.randint(4, 6)  
37     RWT_list.append(goal_RWT)  
38     ceny_za_dil = [1.53, 2.12, 3.40, 1.10, 4.20, 5.10, 0.70]  
39     d_prum_hod_poz = [2142, 2968, 4760, 1540, 5880, 7140, 1960]  
40     cas_na_dodani_materialu = [6, 7, 7, 5, 5, 7, 7]
```

Zahájení „For loopu“ (smyčky) a užití zmíněné výhody generátoru pseudonáhodných čísel. Při každém procházení cykly smyčky je „zamknut“ tzv. „seed“ generátoru a umožňuje reprodukovatelnost simulace v případě jeho nastavení. Ve stávajícím zápisu je nastaveno náhodné číslo, které případný výzkumník může změnit na číslo statické nebo zvolit čísla ze svého definovaného listu. Náhodně zvolená cílová hodnota RWT je dočasně uložena v proměnné „goal_RWT“ a poté vložena do listu „RWT_list“. Tři konečné listy obsahují výchozí hodnoty pro ceny za díl, denní průměrné hodnoty požadavků a čas na dodání materiálu.

KAPITOLA 4. EKONOMICKÉ ZHODNOCENÍ ZAVEDENÍ PFEP

```
41 # Definice proměnlivosti výrobní poptávky po materiálu
42 for y in range(len(d_prum_hod_poz)):
43     d_prum_hod_poz[y] = int(round((d_prum_hod_poz[y] +
44         ↪ d_prum_hod_poz[y]*random.uniform(variabilita_pozadavku*-1, variabilita_pozadavku), 0))
45 # Externí šok na dodací dobu materiálu
46 pocet_soku_cas = (random.choices([1, 2, 3], weights=[7, 3, 2], k=1))
47 for w in range(0, pocet_soku_cas[0]):
48     cas_na_dodani_materialu[random.randrange(0, len(cas_na_dodani_materialu))] =
49     ↪ round((cas_na_dodani_materialu[random.randrange(0, len(cas_na_dodani_materialu))] +
50     ↪ cas_na_dodani_materialu[random.randrange(0,
51     ↪ len(cas_na_dodani_materialu))] * random.uniform(variabilita_casu*-1, variabilita_casu),
52     ↪ 0))
```

Samotné zapsání definice proměnlivosti denní výrobní poptávky po materiálu je v první části úryvku skriptu. „For loop“ (smyčka) prochází list „d_prum_hod_poz“ (Denní průměrná hodnota požadavků) a aplikuje na každou hodnotu náhodnou změnu s uživatelsky zvoleným modifikátorem. Druhá část slouží pro náhodný externí šok časů dodávky materiálu.

```
48 # Model LP
49 # Definice Max/Min
50 prob = LpProblem("RWT", LpMinimize)
51
52 # Definice proměnných
53 sz1 = LpVariable("sz1", lowBound=0, cat='Integer')
54 sz2 = LpVariable("sz2", lowBound=0, cat='Integer')
55 sz3 = LpVariable("sz3", lowBound=0, cat='Integer')
56 sz4 = LpVariable("sz4", lowBound=0, cat='Integer')
57 sz5 = LpVariable("sz5", lowBound=0, cat='Integer')
58 sz6 = LpVariable("sz6", lowBound=0, cat='Integer')
59 sz7 = LpVariable("sz7", lowBound=0, cat='Integer')
60
61 # Definice účelové funkce
62 prob += ceny_za_dil[0]*sz1 + ceny_za_dil[1]*sz2 + ceny_za_dil[2]*sz3 + ceny_za_dil[3]*sz4 +
63     ↪ ceny_za_dil[4]*sz5 + ceny_za_dil[5]*sz6 + ceny_za_dil[6]*sz7
64 # Omezující podmínky
65 prob += ((sz1*ceny_za_dil[0])/d_prum_hod_poz[0]) >= cas_na_dodani_materialu[0]
66 prob += ((sz2*ceny_za_dil[1])/d_prum_hod_poz[1]) >= cas_na_dodani_materialu[1]
67 prob += ((sz3*ceny_za_dil[2])/d_prum_hod_poz[2]) >= cas_na_dodani_materialu[2]
68 prob += ((sz4*ceny_za_dil[3])/d_prum_hod_poz[3]) >= cas_na_dodani_materialu[3]
69 prob += ((sz5*ceny_za_dil[4])/d_prum_hod_poz[4]) >= cas_na_dodani_materialu[4]
70 prob += ((sz6*ceny_za_dil[5])/d_prum_hod_poz[5]) >= cas_na_dodani_materialu[5]
71 prob += ((sz7*ceny_za_dil[6])/d_prum_hod_poz[6]) >= cas_na_dodani_materialu[6]
72 prob += ((ceny_za_dil[0]*sz1 + ceny_za_dil[1]*sz2 + ceny_za_dil[2]*sz3 + ceny_za_dil[3]*sz4 +
73     ↪ ceny_za_dil[4]*sz5 + ceny_za_dil[5]*sz6 + ceny_za_dil[6]*sz7) / sum(d_prum_hod_poz)) >=
74     ↪ goal_RWT
75
76 status = prob.solve()
77 print(LpStatus[status])
78 skore_pocitace.append(int(round(value(prob.objective), 0)))
79 externi_sok_zas = random.choices([random.uniform(0.001, 0.1), random.uniform(0.001, 0.2)],
80     ↪ weights=[10, 3], k=1)
81 skore_pocitace_var.append(int(round((value(prob.objective) +
82     ↪ value(prob.objective)*externi_sok_zas[0]), 0)))
```

KAPITOLA 4. EKONOMICKÉ ZHODNOCENÍ ZAVEDENÍ PFEP

```
78 print(value(prob.objective)/(sum(d_prum_hod_poz)))
79
80 # Hodnoty proměnných
81 for e in range(7):
82     sz_name = "sz" + str(e + 1)
83     LP_promenne.append(value(eval(sz_name)))
```

Definice matematické optimalizační úlohy pro potřeby knihovny „pulp“. Formální zápis modelu můžeme nalézt v 2.3.2 na straně 37. Skóre počítače se rovná optimálnímu řešení účelové funkce, skóre neoptimálního počítače - var se rovná optimálnímu řešení účelové funkce plus náhodný modifikátor.

```
84 # Definice logiky hráče
85 print("Prosím vyplňte průměrnou plánovanou denní skladovou zásobu u každé položky materiálu.")
86 print("Na vyplnění každé hodnoty máte 15s, po uplynutí limitu se nastaví výchozí hodnota.")
87 if automat == "0":
88     input("Pro pokračování stiskněte ENTER")
89 hodnoty_zasob = []
90 potreby = []
91 pokuta = []
92 counter_list = -1
93 for x in prum_sklad_zas:
94     counter_while = 0
95     counter_list += 1
96     while counter_while < 3:
97         print("Cílové globální RWT: " + str(goal_RWT))
98         print("Aktuální globální RWT: " + str(sum(prum_sklad_zas.values())/sum(d_prum_hod_poz)))
99         print("Aktuální RWT materiálu: " + str(prum_sklad_zas[x]/d_prum_hod_poz[counter_list]))
100        print("Dodací doba materiálu: " + str(cas_na_dodani_materialu[counter_list]) + " d")
101        print("Jednotkové náklady na materiál: " + str(ceny_za_dil[counter_list]) + " €")
102        print("Hodnota dodávky: " + str(prum_sklad_zas[x]*ceny_za_dil[counter_list]) + " €")
103        print("\nPočet kusů materiálu - " + x + " : " + str(prum_sklad_zas[x]))
104        print("Materiál: " + str(counter_list + 1) + "/" + str(len(cas_na_dodani_materialu)))
105        print("Období: " + str(i + 1) + "/" + str(pocet_obdobi))
106        if automat == "1":
107            #response = 0
108            response = random.randrange(LP_promenne[counter_list] -
109                                     ↪ int(round(dolni_hrac*LP_promenne[counter_list], 0)), LP_promenne[counter_list]
110                                     ↪ + int(round(horni_hrac*LP_promenne[counter_list], 0)))
109        else:
110            response = pyip.inputInt(timeout=15, min=0, default=prum_sklad_zas[x])
111            prum_sklad_zas[x] = response
112            print("Zadáno: " + str(response))
113            if automat == "0":
114                input("Pokračujte stisknutím ENTER")
115            counter_while += 1
116        hodnoty_zasob.append(prum_sklad_zas[x]*ceny_za_dil[counter_list])
117        potreby.append(d_prum_hod_poz[counter_list]*cas_na_dodani_materialu[counter_list])
118        if (prum_sklad_zas[x]) < LP_promenne[counter_list]:
119            pokuta.append(((cas_na_dodani_materialu[counter_list]*d_prum_hod_poz[counter_list]) -
120                          ↪ prum_sklad_zas[x])*ceny_za_dil[counter_list]*0.5)
120        else:
121            pokuta.append(0)
```

Blok skriptu dedikovaný pro logiku hráče, který na základě proměnné „*automat*“ provede výpočet automaticky, nebo bude uživatel požádán o aktivní participaci. Na ruční zadání hodnot má uživatel patnáct vteřin, které představují v této jednoduché simulaci omezení času pracovního dne.

Zaměřme pozornost na skript od řádku 118, zde je zajištěna aplikace pokuty v případě zadání průměrné denní skladové zásoby menší než optimální denní průměrná skladové zásoba. Při předpokladu by v tomto případě došlo k nedostatku materiálu ve výrobě.

```

122     # Pokuta
123     if sum(hodnoty_zasob) + sum(pokuta) < value(prob.objective):
124         skore_hrace.append(int(round((value(prob.objective) + sum(pokuta)), 0)))
125     else:
126         skore_hrace.append(int(round((sum(hodnoty_zasob) + sum(pokuta)), 0)))
127     sum_hodnoty_zasob.append(sum(hodnoty_zasob))
128     sum_potreby.append(sum(potreby))
129     sum_pokuta.append(sum(pokuta))
130
131     print("--- %s seconds ---" % (time.time() - start_time))
132     print("Skóre hráče: " + '{:,}'.format(sum(skore_hrace)).replace(',', ' '))
133     print("Skóre poč. - var: " + '{:,}'.format(sum(skore_pocitace_var)).replace(',', ' '))
134     print("Skóre počítáče: " + '{:,}'.format(sum(skore_pocitace)).replace(',', ' '))
135     print("Rozdíl: " + '{:,}'.format(sum(skore_hrace) - sum(skore_pocitace)).replace(',', ' '))
136     print("Medián hráče: " + '{:,}'.format(statistics.median(skore_hrace)).replace(',', ' '))
137     print("Medián počítáče - var: " + '{:,}'.format(statistics.median(skore_pocitace_var)).replace(',', ' '))
    ↵

```

Stanovení výše pokuty a vypsání finálních hodnot do příkazové řádky. Řádek 131 vypisuje dobu trvání simulace s pomocí zmíněné knihovny „*time*“.

```

138     # line 1 points
139     x1 = [z for z in range(len(skore_pocitace_var))]
140     y1 = skore_pocitace_var
141     # plotting the line 1 points
142     plt.plot(x1, y1, label = "Skóre poč. - variabilita")
143
144     # line 2 points
145     x2 = [z for z in range(len(skore_pocitace))]
146     y2 = skore_pocitace
147     # plotting the line 2 points
148     plt.plot(x2, y2, label = "Skóre počítáče - ideál")
149
150     # line 3 points
151     x3 = [z for z in range(len(skore_hrace))]
152     y3 = skore_hrace
153     # plotting the line 2 points
154     plt.plot(x3, y3, label = "Skóre hráče")
155
156     # line 4 points
157     x4 = [z for z in range(len(sum_pokuta))]

```



```

158 y4 = sum_pokuta
159 # plotting the line 2 points
160 plt.plot(x4, y4, label = "Pokuta")
161 """
162 # line 5 points
163 x5 = [z for z in range(len(sum_hodnoty_zasob))]
164 y5 = sum_hodnoty_zasob
165 # plotting the line 2 points
166 plt.plot(x5, y5, label = "Hodnoty zásob")
167
168 # line 5 points
169 x6 = [z for z in range(len(sum_potreby))]
170 y6 = sum_potreby
171 # plotting the line 2 points
172 plt.plot(x6, y6, label = "Potřeby")
173 """
174 # naming the x axis
175 plt.xlabel('Počet období')
176 # naming the y axis
177 plt.ylabel('Hodnota v €')
178 # giving a title to my graph
179 plt.title('Výsledná hodnota zásob')
180
181 # show a legend on the plot
182 plt.legend()
183
184 # function to show the plot
185 plt.savefig('chart.png', bbox_inches="tight", dpi=300)
186 plt.show()
187
188 # Příprava seznamu pro export do CSV
189 for i in range(len(skore_hrace)):
190     if i == 0:
191         csv_export.append(["Skore_hrace", "Skore_pocitace_var", "Skore_pocitace"])
192     else:
193         csv_export.append([skore_hrace[i], skore_pocitace_var[i], skore_pocitace[i]])
194
195 with open("skore.csv", "w") as data_file:
196     csv_writer = csv.writer(data_file, delimiter=",")
197     for line in csv_export:
198         csv_writer.writerow(line)

```

Zde využíváme knihovnu „**matplotlib**“ k vytvoření grafu a jeho exportu do složky obsahující skript. Následnou analýzu skóre hráčů lze uskutečnit s podporou exportu dat do souboru „skore.csv“.

4.0.2 Výsledek simulace

Výše popsaný skript nám nyní poslouží k porovnání skóre hráče, který v simulaci reprezentuje scénář **0 - PFEP nezaveden** a skóre počítače představující scénáře **1 - PFEP**

zaveden. Skóre počítače var se odvíjí od optimálního řešení počítače a je k němu přičítán náhodný modifikátor. Simulací provedeme v celkovém počtu pět.

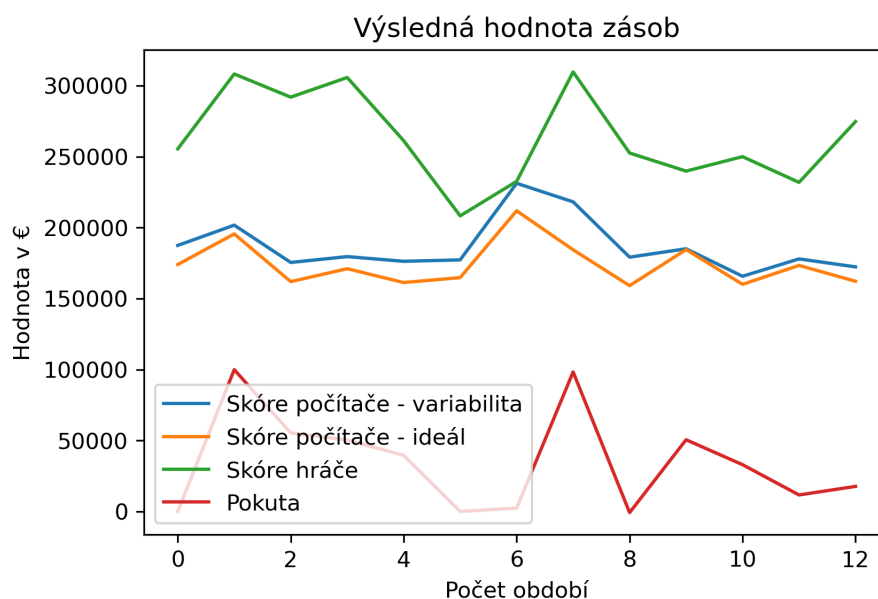
První simulace proběhne s nastavením výchozích hodnot, **druhá** simulace předpokládá nový pracovní nástup. Ve **třetí** simulace je zvýšena variabilita poptávky materiálu, zvýšená variabilita času dodání je poté v simulaci **čtvrté**. V poslední **páté** simulaci jsou zaznamenány hodnoty deseti simulací s nastavenými výchozími hodnotami.

Výchozí hodnoty - 12 období

Parametry simulace:

- Dolní hranice variability automatického hráče - 0.2
- Horní hranice variability automatického hráče - 0.8
- Variabilita poptávky materiálu - 0.1
- Počet období - 12
- Variabilita časů - 0.4

Obrázek 4.1: Výsledný graf simulace - výchozí hodnoty



Zdroj: Vlastní zpracování

Finální skóre hráče je **3 166 910 €**, Skóre poč. - var **2 240 026 €** a skóre počítače je **2 090 120 €**. Dostáváme tedy porovnání rozdílů hodnot scénáře 0 a scénáře 1 s hodnotou **1 076**

KAPITOLA 4. EKONOMICKÉ ZHODNOCENÍ ZAVEDENÍ PFEP

790 €. Z grafu 4.1 na straně 58 je možno vyčíst značný vliv pokuty (Pokuta je aktivována při nižší hodnotě zásob než je optimální množství) na výsledné skóre hráče v prvním a sedmém období.

Tabulka 4.1: Hodnoty simulace - výchozí hodnoty

Období	Skóre hráče	Skóre poč. - var	Skóre počítače
1	308 272	201 752	195 565
2	291 946	175 453	161 957
3	305 782	179 576	171 032
4	261 264	176 263	161 304
5	208 297	177 271	164 781
6	232 601	231 230	211 848
7	309 702	218 162	184 396
8	252 565	179 188	159 092
9	239 831	185 098	184 517
10	250 014	165 735	160 048
11	231 885	177 951	173 376
12	274 751	172 347	162 204

Zdroj: Vlastní zpracování

Nezkušený pracovník - 12 období

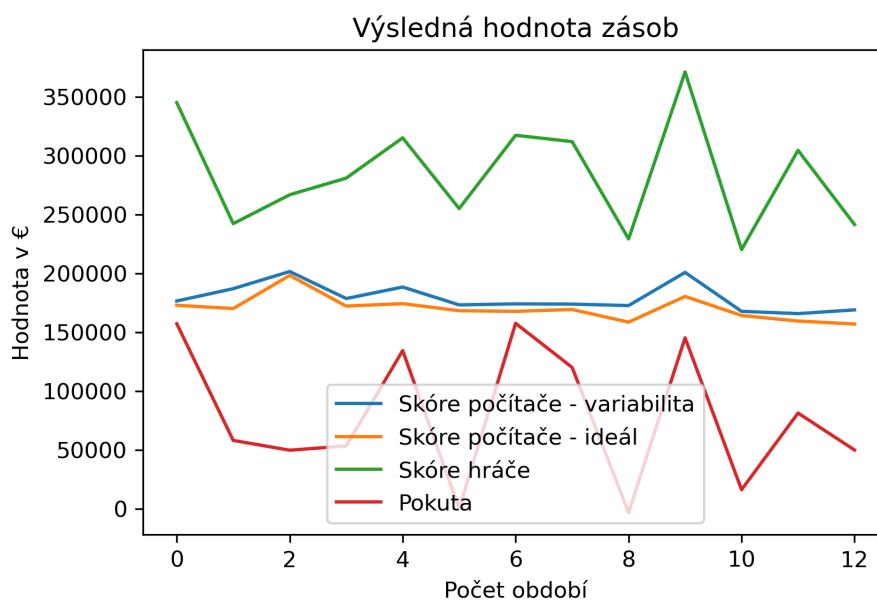
Tento scénář 0 simuluje stav při užití nového a nezkušeného pracovníka do role disponenta pro linku dveřního pantu za nepoužití PFEP. Scénář 1 poté pracuje s implementací PFEP.

Parametry simulace:

- Dolní hranice variability automatického hráče - 0.4
- Horní hranice variability automatického hráče - 0.9
- Variabilita poptávky materiálu - 0.1
- Počet období - 12
- Variabilita časů - 0.4

Skóre hráče v druhé simulaci se oproti první navýšilo na hodnotu **3 356 681 €**. Opačný trend můžeme pozorovat u počítače - var s hodnotou **2 152 494 €** a počítače s finálním skóre **2 039 508 €**. Rozdíl skóre hráče a počítače je **1 317 173 €**. Graf 4.2 na straně 60 ukazuje vliv změny parametrů na hodnoty pokuty.

Obrázek 4.2: Výsledný graf simulace - nezkušený pracovník



Zdroj: Vlastní zpracování

Tabulka 4.2: Hodnoty simulace - nezkušený pracovník

Období	Skóre hráče	Skóre počítače	Skóre poč. - var
1	242 282	186 978	170 094
2	266 823	201 575	198 236
3	280 988	178 601	172 173
4	315 216	188 372	174 273
5	255 099	173 189	168 252
6	317 404	174 065	167 700
7	312 005	173 845	169 303
8	229 263	172 622	158 539
9	371 249	200 798	180 383
10	220 208	167 718	164 177
11	304 596	165 771	159 451
12	241 548	168 960	156 927

Zdroj: Vlastní zpracování

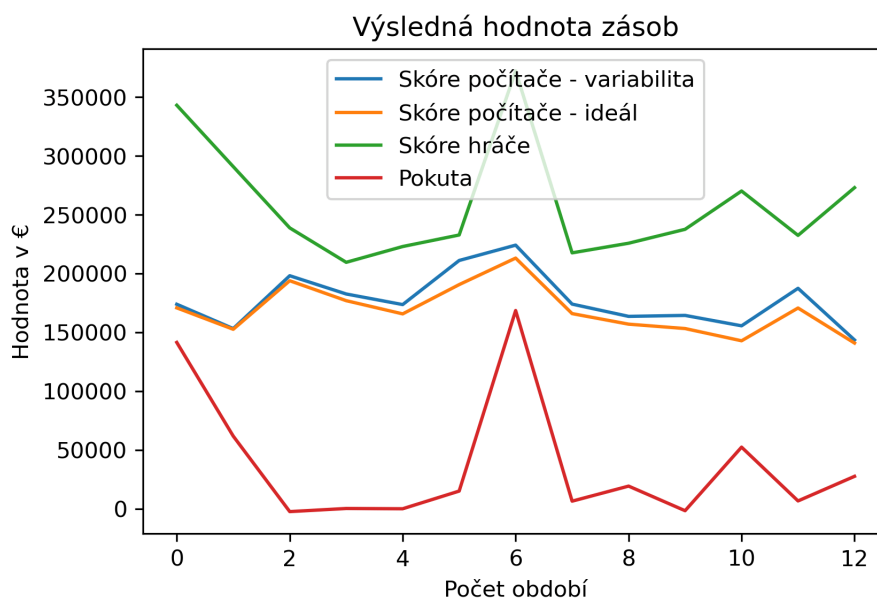
Variabilita poptávky materiálu - 12 období

Změny parametrů v této simulaci ovlivňují poptávku po materiálu, která je díky navýšení modifikátorů více proměnlivá. Parametry simulace:

- Dolní hranice variability automatického hráče - 0.4
- Horní hranice variability automatického hráče - 0.9
- Variabilita poptávky materiálu - 0.4
- Počet období - 12

- Variabilita časů - 0.4

Obrázek 4.3: Výsledný graf simulace - variabilita poptávky materiálu



Zdroj: Vlastní zpracování

Třetí simulace v pořadí obsahuje skóre pro hráče s hodnotou **3 024 628 €**, počítač - var má skóre **2 131 853 €** a skóre počítače nabývá hodnoty **2 023 831 €**. Z prozatímních simulací představují tyto hodnoty nejmenší rozdíl scénáře 0 a scénáře 1 s číslem **1 000 797 €**. V

Tabulka 4.3: Hodnoty simulace - variabilita poptávky materiálu

Období	Skóre hráče	Skóre počítače	Skóre poč. - var
1	290 983	153 246	152 632
2	238 873	198 188	193 925
3	209 597	182 669	176 971
4	223 019	173 648	165 734
5	232 836	211 152	190 633
6	372 352	224 203	213 193
7	217 648	174 053	165 967
8	225 827	163 623	157 001
9	237 668	164 442	153 265
10	270 236	155 589	142 856
11	232 507	187 488	170 731
12	273 082	143 552	140 923

Zdroj: Vlastní zpracování

grafu 4.3 na straně 61 si můžeme všimnout značného efektu pokuty v období 0 a 6.

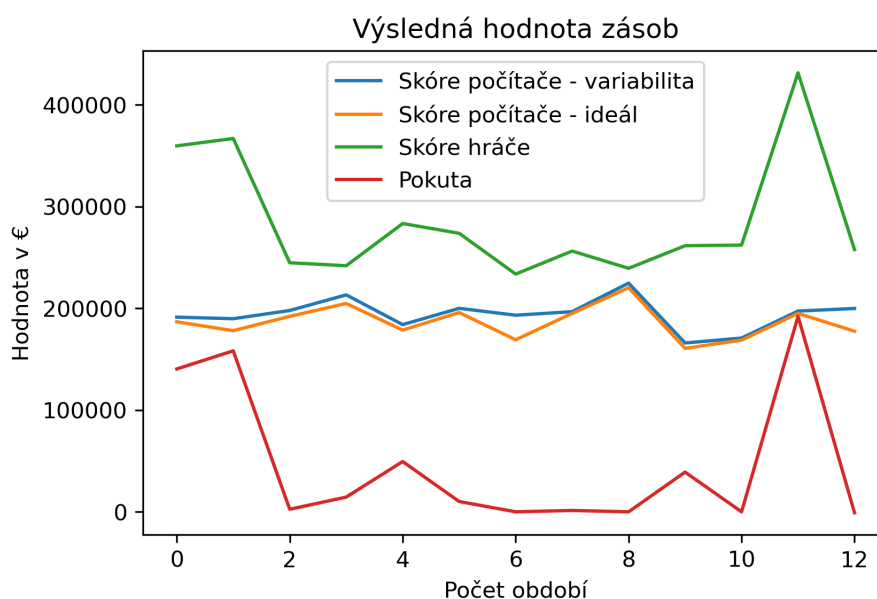
Variabilita časů - 12 období

Parametry simulace:

- Dolní hranice variability automatického hráče - 0.4
- Horní hranice variability automatického hráče - 0.9
- Variabilita poptávky materiálu - 0.1
- Počet období - 12
- Variabilita časů - 0.8

Z předposlední simulace jsme získali finální skóre hráče s hodnotou **3 353 292 €**, počítač var **2 333 888 €** a nakonec počítač se skóre **2 236 033 €**. Rozdíl v tomto případě činí **1 117 259 €**.

Obrázek 4.4: Výsledný graf simulace - variabilita času na dodání materiálu



Zdroj: Vlastní zpracování

Graf 4.4 na straně 62 obsahuje značný nárůst pokuty v období jedenáctém, která dosahuje výše skóre hráče.

KAPITOLA 4. EKONOMICKÉ ZHODNOCENÍ ZAVEDENÍ PFEP

Tabulka 4.4: Hodnoty simulace - variabilita času na dodání materiálu

Období	Skóre hráče	Skóre počítače	Skóre poč. - var
1	366 970	189 824	178 056
2	244 833	197 897	192 165
3	241 910	213 272	204 800
4	283 388	184 013	178 622
5	273 792	200 035	195 905
6	233 698	193 261	169 115
7	256 310	196 748	195 060
8	239 361	224 811	220 295
9	261 562	165 975	160 686
10	262 206	170 791	168 759
11	431 522	197 372	195 032
12	257 740	199 889	177 538

Zdroj: Vlastní zpracování

Tabulka 4.5: Hodnoty simulace - sumy 10 simulací

Simulace	Skóre hráče	Skóre počítače	Skóre poč. - var	Úspora hráč - poč.	Úspora v %
1	3 820 934	2 329 122	2 212 441	1 608 493	0,43
2	3 616 522	2 388 213	2 247 864	1 368 658	0,38
3	3 652 381	2 278 949	2 155 735	1 496 646	0,41
4	3 397 918	2 301 925	2 211 343	1 186 575	0,35
5	3 592 218	2 274 383	2 170 674	1 421 544	0,4
6	3 406 393	2 327 527	2 139 591	1 266 802	0,38
7	3 606 293	2 362 570	2 215 227	1 391 066	0,39
8	3 330 568	2 368 853	2 206 155	1 124 413	0,34
9	2 961 019	2 278 587	2 156 298	804 721	0,28
10	3 483 350	2 292 496	2 164 310	1 319 040	0,38

Zdroj: Vlastní zpracování, Data:

<https://or.justice.cz/ias/ui/vypis-sl-firma?subjektId=151117>

Výchozí hodnoty - 12 období / 10 simulací

4.0.3 Porovnání variant řešení PFEP

Předešlá simulace zavedení PFEP nám poukázala na možné benefity zavedení a používání nástroje v oblasti optimalizace zásob s vlivem na RWT. Tvorba nástroje v Excelu není jediná cesta k cíli, ale máme několik možností zmíněných v kapitole 2.3.1 na straně 29.

Proplanner

Komplexní profesionální řešení obsahující nejen kolaborativní PFEP nástroj, ale také eKanban, eKitting a FlowPlanner. Řešení založené na databázi a grafické nadstavbě softwaru Proplanner. Specifika tohoto řešení jsou široká škála integrace s ostatními systémy

zahrnující nejen ERP, ale i PLM a jiné inženýrské programy zahrnující zmíněné nástroje Proplanner. Přímé napojení na eKanban pomáhá se správou Kanban karet a jejich sledování. Komunikace s eKitting umožňuje automatickou tvorbu zásobovacích balíčků, neboli tzv. „kittů“ pro interní zásobování linek. Korektní nastavení adres „supermarketů“, neboli výdejních bodů pro výrobní linky a samotných montážních linek umožní snadnou tvorbu studií pohybu materiálu a lidí s výstupným Sankey diagramem. [29]

loopPFEP

Nástroj založený na centrální databázi s grafickým pohledem postaveným na webových technologiích. Z tohoto důvodu není nutná instalace nástroje a je možné k němu přistupovat i z mobilního zařízení. Mezi hlavní publikované výhody patří:

- Dynamická workflow
- API pro snadnou komunikaci s externím světem
- PFEP analytika
- Propojení s externími partnery
- Extenzivní školení

Náklady na řešení

Předpokládané hrubé náklady na řešení získáme z dostupných zdrojů komerčního softwaru loopPFEP a Proplanner. Hrubé náklady na zavedení interního řešení získáme odhadem pracnosti základního řešení a řešení pokročilého. I přesto, že se jedná o interní řešení, uvažujeme náklady na nutné najmutí externích IT odborníků na přizpůsobení Excel nástroje.

Proplanner nabízí své řešení v podobě Software as a Service (SaaS) v balíčcích od pětadvaceti uživatelů. Pro naše potřeby uvažujeme základní balík s pětadvaceti uživateli, který je nabízen za 200 USD na uživatele za měsíc. Při aktuálním kurzu¹ se tato částka rovná 5 020 CZK na uživatele za měsíc. Konečná částka činí **1 506 000 CZK za rok**.

¹Aktuální kurz 1 USD = 25,1

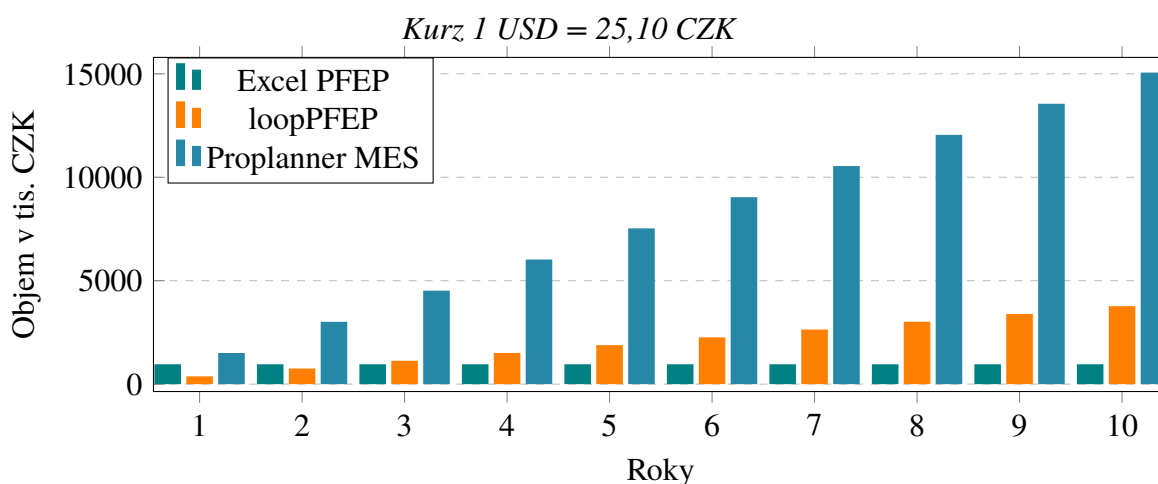
LoopPFEP pracuje také s modelem SaaS a nabízí předplatné v hodnotě 50 USD na uživatele za měsíc. Při současném kurzu se toto rovná 1 255 CZK na uživatele za měsíc. Celková částka pro pětadvacet uživatelů za rok činí **376 500 CZK**.

Excel PFEP náklady vypočítáme z náročnosti na implementaci a odhadovaných hrubých nákladů na externí konzultanty. Hrubá odhadovaná časová náročnost na implementaci základního řešení Excel PFEP je čtyři měsíce. Pro úspěšné dokončení budeme potřebovat externího IT konzultanta/technika pro naprogramování základních skript, který bude vyplacen částkou 120 000 CZK. Pokročilé řešení bude pokrývat větší oblast a také obsahovat složitější simulace a výpočty. Hrubá odhadovaná časová náročnost na implementaci je jeden rok. K náročnějšímu řešení bude potřeba navýšení externích sil o jednoho člena a celková výše kompenzace konzultantů naroste na 960 000 CZK.

Do výsledné kalkulace nezapočítáváme náklady na dedikovaného PFEP manažera, protože figuruje ve všech scénářích. Porovnání **kumulativních** nákladů v čase zobrazuje graf 4.5 na straně 65. Porovnává se zde komerční řešení a pokročilá verze Excel PFEP, jejíž kumulativní náklady zůstávají konstantní z důvodu absence předplácení za nástroj. Vidíme, že už třetí rok je nástroj loopPFEP v celkové výši nákladů dražší než Excel PFEP.

[23, 29]

Obrázek 4.5: Porovnání kumulativních nákladů na jednotlivá řešení PFEP



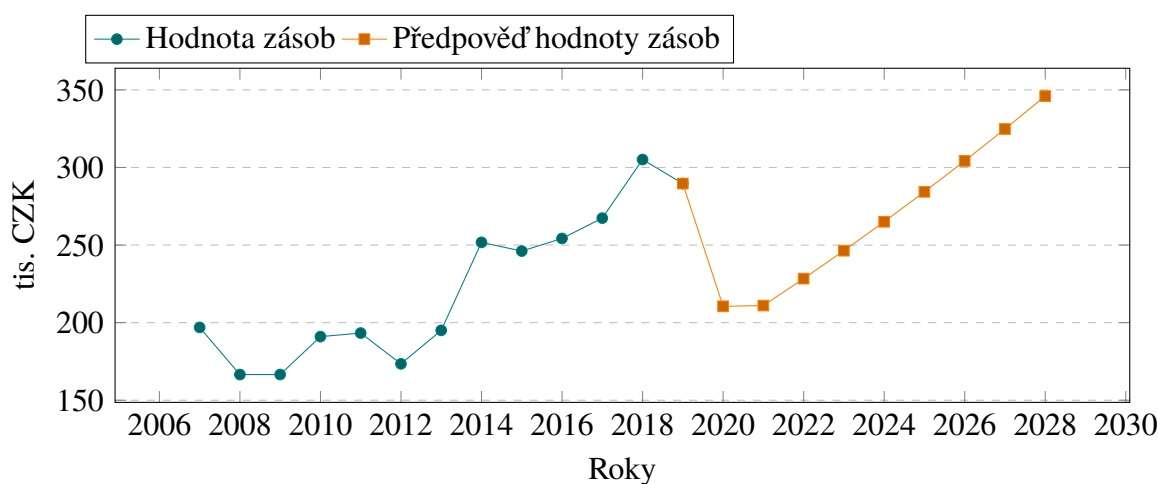
Zdroj: Vlastní zpracování, data: <https://www.proplanner.com/pricing> a <https://www.capterra.com/p/157775/loopPFEP/>

Efektivita vynaložených prostředků

K finálnímu rozhodnutí o cestě k PFEP je také nutné mít informaci o efektivitě vynaložených finančních prostředků na potenciální úspory. Tuto informaci získáme porovnáním nákladů nástroje PFEP s hodnotou potenciální úspory po implementaci PFEP.

K odhadnutí budoucí hodnoty zásob použijeme data z výročních zpráv Witte, která poslouží k predikci budoucí hodnoty zásob. V roce 2020 představíme externí negativní šok, který poníží hodnotu zásob o 30%². Tímto simulujeme dopady utlumení prodeje a výroby automobilů z důvodu pandemie Covid-19. Na výsledném grafu 4.6 na straně 66 můžeme vidět historickou hodnotu zásob ve Witte a předpověď budoucích hodnot.

Obrázek 4.6: Podíl zásob na celkových aktivech Witte



Zdroj: Vlastní zpracování, data:

<https://or.justice.cz/ias/ui/vypis-sl-firma?subjektId=151117>

Nyní potřebujeme znát hodnotu dosažitelné úspory zásob po zavedení nástroje PFEP. Z našich simulací a tabulky 4.5 na straně 63 můžeme vyčíst průměrnou úsporu 37,4%³. Konzervativní odhad horní hranice z důvodu neobsáhnutí veškerých proměnných v simulaci stanovíme na 30%⁴, střední hodnotu na 20% a spodní hranici na 10%. Uvažujeme stejnou účinnost na snížení zásob u všech nástrojů.

²Částku 30% přebíráme ze středního scénáře dle [21].

³Simulace nezahrnují veškeré proměnné vstupující do možných úspor.

⁴Toto číslo koresponduje s mnohými studiemi z reálných implementací PFEP.

KAPITOLA 4. EKONOMICKÉ ZHODNOCENÍ ZAVEDENÍ PFEP

Výsledek analýzy je zaznamenán v tabulkách 4.6, 4.7 a 4.8 na straně 67. Třetí sloupec tabulek vyjadřuje predikované úspory dle stanovené hodnoty 10%, 20% a 30%. Čtvrtý až sedmý sloupec vyobrazuje poměr nákladů na nástroj a úspor, který je v procentech.

Tabulka 4.6: Efektivita vynaložených finančních prostředků - spodní varianta 10%

Čítatel (náklady na nástroj) poměru je kumulativní hodnota

Roky	Zás. v tis. CZK	Úsp. v tis. CZK	Excel v %	Excel - pok. v %	loopPFEP v %	Proplanner v %
2019	289 639	28 964	0,415	3,315	1,302	5,2
2020	210 539	21 054	0,57	4,56	3,582	14,307
2021	211 015	21 102	0,569	4,55	5,36	21,411
2022	228 374	22 838	0,526	4,204	6,604	26,378
2023	246 373	24 638	0,488	3,897	7,651	30,563
2024	265 014	26 502	0,453	3,623	8,536	34,096
2025	284 298	28 430	0,423	3,377	9,283	37,081
2026	304 222	30 423	0,395	3,156	9,914	39,602
2027	324 787	32 479	0,37	2,956	10,447	41,732
2028	345 994	34 600	0,347	2,775	10,896	43,527
Celkem	2 710 255	271 030	0,045	0,355	7,651	30,562

Zdroj: Vlastní zpracování, Data:

<https://or.justice.cz/ias/ui/vypis-sl-firma?subjektId=151117>

Tabulka 4.7: Efektivita vynaložených finančních prostředků - střední varianta 20%

Čítatel (náklady na nástroj) poměru je kumulativní hodnota

Roky	Zás. v tis. CZK	Úsp. v tis. CZK	Excel v %	Excel - pok. v %	loopPFEP v %	Proplanner v %
2019	289 639	57 928	0,208	1,658	0,651	2,6
2020	210 539	42 108	0,285	2,28	1,791	7,154
2021	211 015	42 203	0,285	2,275	2,68	10,706
2022	228 374	45 675	0,263	2,102	3,302	13,189
2023	246 373	49 275	0,244	1,949	3,826	15,282
2024	265 014	53 003	0,227	1,812	4,268	17,049
2025	284 298	56 860	0,212	1,689	4,642	18,541
2026	304 222	60 845	0,198	1,578	4,957	19,802
2027	324 787	64 958	0,185	1,478	5,224	20,866
2028	345 994	69 199	0,174	1,388	5,449	21,764
Celkem	2 710 255	542 054	0,023	0,178	3,826	15,281

Zdroj: Vlastní zpracování, Data:

<https://or.justice.cz/ias/ui/vypis-sl-firma?subjektId=151117>

Tato hodnota ve své podstatě vyjadřuje kolik firma musela vynaložit finančních prostředků na 1 CZK úspory. Například hodnota pro Proplanner v první variantě a roce 2019 5,2% znamená vynaložení 0,052 CZK na 1 CZK úspor. Náklady v následných letech jsou pak kumulativní⁵.

Výsledky naznačují příznivější zhodnocení vynaložených prostředků do nástrojů vytvořených v Excelu oproti komerčnímu softwaru. Pro finální verdikt by však bylo nutno

⁵Aktuální poměr nákladů/úspor v každém roce bude oscilovat kolem hodnoty roku 2019.

KAPITOLA 4. EKONOMICKÉ ZHODNOCENÍ ZAVEDENÍ PFEP

Tabulka 4.8: Efektivita vynaložených finančních prostředků - horní varianta 30%

Čítatel (náklady na nástroj) poměru je kumulativní hodnota

Roky	Zás. v tis. CZK	Úsp. v tis. CZK	Excel v %	Excel - pok. v %	loopPFEP v %	Proplanner v %
2019	289 639	86 892	0,139	1,105	0,434	1,734
2020	210 539	63 162	0,19	1,52	1,194	4,769
2021	211 015	63 305	0,19	1,517	1,787	7,137
2022	228 374	68 513	0,176	1,402	2,202	8,793
2023	246 373	73 912	0,163	1,299	2,551	10,188
2024	265 014	79 505	0,151	1,208	2,846	11,366
2025	284 298	85 290	0,141	1,126	3,095	12,361
2026	304 222	91 267	0,132	1,052	3,305	13,201
2027	324 787	97 437	0,124	0,986	3,483	13,911
2028	345 994	103 799	0,116	0,925	3,633	14,509
Celkem	2 710 255	813 082	0,015	0,119	2,551	10,188

Zdroj: Vlastní zpracování, Data:

<https://or.justice.cz/ias/ui/vypis-sl-firma?subjektId=151117>

vypracovat důkladnou analýzu, která by obsáhla více proměnných a prozkoumala detailně potřeby firmy. Nástroj s nejhorším zhodnocením vložených finančních prostředků má také nejširší portfolio funkcí a je připraven na komunikaci s nemalým množstvím ostatních softwarových řešení.

Závěr

Diplomová práce se zabývala specifickou oblastí Lean manufacturing zvanou Plan for every part (PFEP). Jedná se o velmi důležitou komponentu Toyota production system (TPS) a Lean manufacturing, která je však velmi málo probádána a často opomíjena při implementaci „štíhlých“ postupů ve firmách. První část práce seznamuje čtenáře s výrobcem automobilových komponent WITTE Automotive (Witte), který poskytl autorovi potřebná data a možnost konzultací s interními odborníky oddělení logistiky. Základní představení firmy je následováno popisem procesů řízení a optimalizace zásob Witte a představení hlavního celopodnikového Key Performance Indicator (KPI) Reichweite (RWT).

Druhá část práce krátce seznamuje čtenáře s historií TPS a poskytuje tak kontext pro pochopení filozofie TPS. Dále se kapitola věnuje otázkám plýtvání ve výrobě a motivace pro výstavbu PFEP nástroje. Podstatná část druhé kapitoly je zaměřena na návrh nástroje v prostředí tabulkového kalkulátoru Microsoft Excel. Autor rozebírá strukturu dat nástroje a poté podrobněji zkoumá dvě oblasti optimalizace zásob. První oblastí je právě simulace zmíněného RWT, ve které autor navrhuje řešení s pomocí nástrojů matematického programování. Druhou oblast optimalizace počtu kusů materiálu v obalu řeší autor prostřednictvím Visual Basic for Applications (VBA) skriptu a porovnává své řešení s možným řešením pomocí nelineárního matematického modelu.

Třetí část se věnuje samotným krokům implementace nástroje PFEP. Postupné kroky implementace jsou doplněny nutnými technickými a uživatelskými předpoklady. Je zde kladen důraz na důležitost manažera PFEP jakožto klíčový prvek v implementaci a životaschopnosti nástroje.

V poslední kapitole je věnován prostor jednoduché simulaci vytvořené ve skriptovacím jazyce Python, která poukazuje na benefity PFEP v rámci úspory zásob. Prvotní rozbor kódu je následován zhodnocením výsledných dat pěti simulací. Dosažené teoretické hodnoty úspor jsou konfrontovány s náklady na tři, respektive čtyři možné přístupy k výstavbě nástroje PFEP. Dosažené výsledky z konečné analýzy ukazují lepší zhodnocení vynaložených finančních prostředků do Excel PFEP nástroje oproti komerčnímu řešení.

Předložené řešení nástroje PFEP slouží jako úvodní, ale funkční přestavení možností PFEP. Díky své flexibilitě poskytuje potenciál pro rozpracování poznatků této práce o další aspekty firemních procesů. Údaje o rozměrech obalů mohou sloužit pro výpočty nutné skladovací a výrobní plochy. Spojení hodnocení dodavatelů s ABC analýzou odhalí potenciální komplikace v případě klíčových výrobků, které dodává nespolehlivý dodavatel.

Dále je zde potenciál pro rozvoj problematiky v akademické sféře. V současné době neexistuje publikace psaná v českém jazyce, která by se PFEP věnovala. Podobná situace je i v případě absolventských prací psaných v českém jazyce. Server theses.cz při zadání textového řetězce "Plan for every part" eviduje dvě diplomové a jednu bakalářskou práci, které obsahují zmínku o tomto nástroji, ale nezaobírají se PFEP v náležitém detailu. Akademická sociální síť researchgate.net neeviduje žádné články o PFEP z rukou českých akademiků.

Seznam použité literatury

Tištěná média

- [1] ŠIMONOVÁ, Stanislava. *Databázové systémy I*. Pardubice: Univerzita Pardubice, 2013. ISBN: 978-80-7395-702-5.
- [2] CONRAD, Tim a Robyn ROOKS. *Turbo flow: using plan for every part (PFEP) to turbo charge your supply chain*. Boca Raton, Florida, USA: CRC Press, 2010. ISBN: 9781439820674.
- [3] Harris, R. a Harris, C. *Making materials flow: a lean material-handling guide for operations, production-control, and engineering professionals*. A lean toolkit method and workbook. Lean Enterprise Institute, 2003. ISBN: 9780974182490.
- [4] JOHNSON P. Fraser, Michiel R. LEENDERS a Anna E. FLYNN. *Purchasing and supply management*. 14th. McGraw-Hill/Irwin, 2011. ISBN: 978-0-07-337789-6.
- [5] KRATOCHVÍLOVÁ, Lucie. *Řízení zásob - Pojistné zásoby, karenční dny, dosah zásob, profil doby dosahu, LSF, manuální úprava plánu dodávek*. Interní materiály Witte. 2017.
- [6] KRATOCHVÍLOVÁ, Lucie. *Optimalizace zásob - Dispozice – RWT, ZANBEST*. Interní materiály Witte. 2017.
- [7] MAGAL, Simha a Jeffrey WORD. *Integrated business processes with ERP systems*. Hoboken, NJ: Wiley, 2012. ISBN: 0470478446.
- [8] Namkoong, Young. “Impact of the Zaibatsu on Japan’s Political Economy: Pre and Post War Period”. In: *International Area Review* 9.2 (2006), s. 1–22. DOI: 10.1177/223386590600900207.
- [9] “PFEP-Oriented In-Plant Logistics Planning Method for Assembly Plants.” In: *ICLEM 2010* (2010), s. 1396–1404. DOI: 10.1061/41139(387)192. URL: <http://ascelibrary.org/doi/10.1061/41139%5C%28387%5C%29192>.
- [10] PLEVNÝ, Miroslav a Miroslav ŽIŽKA. *Modelování a optimalizace v manažerském rozhodování*. Západočeská univerzita v Plzni, 2010. ISBN: 978-80-7043-933-3.

- [11] Ross, D.F. *Introduction to Supply Chain Management Technologies*. Resource Management. CRC Press, 2016. ISBN: 9781439837535.
- [12] Smalley, A. *Creating Level Pull: A Lean Production-system Improvement Guide for Production-control, Operations, and Engineering Professionals*. Lean Tool Kit. Lean Enterprise Institute, 2004. ISBN: 9780974322506.
- [13] Wagner, B. a Enzler, S. *Material Flow Management: Improving Cost Efficiency and Environmental Performance*. Sustainability and Innovation. Physica-Verlag HD, 2005. ISBN: 9783790816655.
- [14] Womack, J. a Shook, J. *Gemba Walks, 2nd Ed.: Expanded 2nd Edition*. Lean Enterprise Institute, Incorporated, 2013. ISBN: 9781934109458. URL: <https://books.google.cz/books?id=wXsnAQAAQBAJ>.

Online média

- [15] *Štíhlé principy a procesně orientovaná výroba*. URL: <https://www.systemonline.cz/rizeni-vyroby/stihle-principy-a-procesne-orientovana-vyroba.htm> (cit. 12. 03. 2020).
- [16] *ABOUT US*. URL: <https://www.strattec.com/about> (cit. 08. 02. 2020).
- [17] *ADAC'S DNA*. URL: <https://www.adacautomotive.com/about-adac/> (cit. 07. 02. 2020).
- [18] *Historie - WITTE Automotive*. URL: <https://www.witte-automotive.cz/live/Historie/historie.aspx> (cit. 01. 12. 2019).
- [19] *How Do ERP Systems Work?* URL: <https://www.selecthub.com/enterprise-resource-planning/how-do-erp-systems-work/> (cit. 16. 02. 2020).
- [20] HUWART, Jean-Yves a VERDIER, LOÏC. *Economic globalisation: origins and consequences*. 2013. URL: http://www.oecd-ilibrary.org/economics/economic-globalisation_9789264111905-en.

- [21] *Infographic: Forecasting Covid-19's Effect on the Auto Industry*. URL: <https://www.bain.com/insights/forecasting-covid-19-effect-on-the-auto-industry-infographic/> (cit. 08. 05. 2020).
- [22] *Lokální i globální trendy ERP pro rok 2015*. URL: <https://www.systemonline.cz/erp/lokalni-i-globalni-trendy-erp-pro-rok-2015.htm> (cit. 24. 02. 2020).
- [23] *loopPFEP*. URL: <https://loopsupplysystems.com/pfep/> (cit. 20. 03. 2020).
- [24] *Milióntá přepravka odbavena z WITTE Logistického centra*. URL: https://www.witte-automotive.cz/Mili%C3%B3nt%C3%A1-p%C5%99pravka-odbavena-z-WITTE-Logistick%C3%A9ho-centra_430.aspx (cit. 13. 02. 2020).
- [25] *Milk Run, aneb vychytané dodávky materiálu*. URL: https://www.witte-automotive.cz/Milk-Run--aneb-vychytan%C3%A9-dod%C3%A1vky-materi%C3%A1lu_170.aspx (cit. 11. 02. 2020).
- [26] *Náběh štíhlé logistiky v nové výrobě autodílů*. URL: <https://logistika.ihned.cz/c1-65536730-nabeh-stihle-logistiky-v-nove-vyrobe-autodilu> (cit. 13. 02. 2020).
- [27] *Our clients*. URL: <https://www.witte-automotive.com/live/References-customers/references.aspx> (cit. 04. 02. 2020).
- [28] *Plan for Every Part Keeps Your ERP at its Best*. URL: <https://www.leandna.com/blog/tuneup-erp-plan-every-part-pfep/> (cit. 16. 02. 2020).
- [29] *Plan For Every Part (PFEP)*. URL: <https://www.proplanner.com/solutions/material-logistics-planning/plan-for-every-part/> (cit. 20. 03. 2020).
- [30] *Relational database*. URL: <https://www.britannica.com/technology/relational-database> (cit. 17. 02. 2020).
- [31] *Software | IndustryStar Solutions*. URL: <https://www.industrystarsolutions.com/software/> (cit. 20. 03. 2020).

- [32] *Spreadsheet | computing*. URL: <https://www.britannica.com/technology/spreadsheet> (cit. 19. 02. 2020).
- [33] *SQL Data Types*. URL: <https://www.journaldev.com/16774/sql-data-types> (cit. 16. 02. 2020).
- [34] *SQL Syntax*. URL: https://www.w3schools.com/sql/sql_syntax.asp (cit. 07. 02. 2020).
- [35] *V roce 1899 byl založen jako rodinný podnik, dnes slaví WITTE Automotive 120 let - WITTE Automotive*. URL: https://www.witte-automotive.cz/V-roce-1899-byl-zalo%C5%BEen-jako-rodinn%C3%BD-podnik--dnes-slav%C3%AD-WITTE-Automotive-120-let_1818.aspx (cit. 01. 12. 2019).
- [36] *VAST - Vehicle Access Systems Technology*. URL: <https://www.vastglobal.com/> (cit. 08. 02. 2020).
- [37] *Veřejný rejstřík a Sbírka listin - Ministerstvo spravedlnosti České republiky*. [Údaje platné ke dni 1.12.2019]. URL: <https://or.justice.cz/ias/ui/rejstrik-firma.vysledky?subjektId=151117&typ=PLATNY> (cit. 01. 12. 2019).
- [38] *Výrobky - WITTE Automotive*. URL: <https://www.witte-automotive.cz/live/vyrobky/vyrobky.aspx> (cit. 21. 03. 2020).
- [39] *WITTE ACCESS TECHNOLOGY, s.r.o.* URL: <https://www.witte-automotive.cz/live/WITTE-ACCESS-TECHNOLOGY-s-r-o/witte-ostrov.aspx> (cit. 03. 02. 2020).
- [40] *WITTE Milk run jen se štíhlými dodavateli - WITTE Automotive*. URL: https://www.witte-automotive.cz/WITTE-Milk-run-jen-se-%C5%A1t%C3%ADhl%C3%BDmi-dodavateli_54.aspx (cit. 10. 02. 2020).

Seznam obrázků

1.1	Historie Witte	8
1.2	Výrobky Witte	10
1.3	Výrobky z řady kapotových konceptů	11
1.4	System WITOL®	12
1.5	Podíl zásob na celkových aktivech Witte	14
1.6	Logistické náklady v kalkulacích	15
1.7	Příklad měsíčního reportingu RWT	18
1.8	Příklad denního reportingu RWT	19
1.9	Změna přístupu v SCM řízení	20
1.10	Proces simulace optimalizace RWT	22
2.1	Porovnání klasického pojetí cenotvorby s Lean pojetím	25
2.2	Magický kvadrant ERP systémy - střední, velké a nadnárodní korporace	27
2.3	Uživatelské rozhraní Proplanner PFEP	30
2.4	Přední dveřní pant	31
2.5	Uživatelské prostředí nástroje	32
2.6	Správce názvů	33
2.7	Návrh nového procesu simulace optimalizace RWT	35
2.8	Výsledek optimalizace pomocí OpenSolver	38
2.9	NP model optimalizace počtu kusů materiálu v L2 obalu	44
3.1	TPS + PFEP	48
3.2	Tok dat pro potřeby PFEP	49
4.1	Výsledný graf simulace - výchozí hodnoty	58

4.2	Výsledný graf simulace - nezkušený pracovník	60
4.3	Výsledný graf simulace - variabilita poptávky materiálu	61
4.4	Výsledný graf simulace - variabilita času na dodání materiálu	62
4.5	Porovnání kumulativních nákladů na jednotlivá řešení PFEP	65
4.6	Podíl zásob na celkových aktivech Witte	66

Seznam tabulek

2.1	Přehled podniků používající CRM a ERP systémy v ČR	26
2.2	Hodnoty modelu	37
2.3	Porovnání exekuce VBA skriptu a Řešitele	45
4.1	Hodnoty simulace - výchozí hodnoty	59
4.2	Hodnoty simulace - nezkušený pracovník	60
4.3	Hodnoty simulace - variabilita poptávky materiálu	61
4.4	Hodnoty simulace - variabilita času na dodání materiálu	63
4.5	Hodnoty simulace - sumy 10 simulací	63
4.6	Efektivita vynaložených finančních prostředků - spodní varianta 10% . . .	67
4.7	Efektivita vynaložených finančních prostředků - střední varianta 20% . . .	67
4.8	Efektivita vynaložených finančních prostředků - horní varianta 30% . . .	68

Slovník

5s	Pět základních pravidel - rozděl (sort), uspořádej (straighten), uklid' (systematic cleaning), standardizuj (standardise) a dodržuj (sustain). 46
ABC	ABC analýza. Vychází z Paretova pravidla a rozděluje položky do tří kategorií ABC podle jejich podílu na celkové hodnotě parametru. . 18, 32, 70
COGI	SAP transakce používající se pro korekci chyb při automatických transakcích spojených s materiálem atd.. 20
Just in time	Moderní zásobovací systém s cílem dodávat záskyby „včas“ a minimalizovat tak skladové prostory. 23, 24, 37
Kaizen	Filozofie kontinuálního zlepšování postupů a procesů. Využívá se zejména ve strojírenství a průmyslu.. 24, 46
Kanban	Systém pro efektivní zásobování výrobních společností. Inspirace v amerických supermarketech. 46, 63, 64
L2 obal	Level 2 obal - typicky krabice a jiné. 30, 39, 40
Lean manufacturing	Moderní filozofie řízení podniku na bázi minimalizace plýtvání. 23, 24, 69
LSF	Mimořádný požadavek na materiál. 17, 32
Muda	Původem japonské slovo označující plýtvání a ztráty. 46
Mura	Původem japonské slovo označující nevyváženost a nepravidelnost. 46
Muri	Původem japonské slovo označující neopodstatněné přetěžování.. 46
SAP	Firma, která vyvíjí stejnojmenný ERP software. 16, 19, 26, 35

Supply chain management Řízení dodavatelského řetězce. 15

Seznam zkratek

BI	Business Intelligence. 33
Brose	Brose Fahrzeugteile GmbH
Co. KG.	12
BT	Business Team. 17–20
BW	Business warehouse. 19
CRM	Customer Relationship Management. 26, 77
CZK	Koruna česká. 7, 64, 65, 67
DCL	Data Control Language. 28
DDL	Data Definition Language. 28
DML	Data Manipulation Language. 28
ERP	Enterprise resource planning. 26, 31, 32, 49, 64, 77
FIFO	First in first out. 14
Ford	Ford Motor Company. 12, 23
IT	Informační technologie. 64, 65
KPI	Key Performance Indicator. 7, 17, 51, 69
Lamborghini	Automobili Lamborghini S.p.A.. 12
LP	Lineární programování. 36
MAN	MAN Nutzfahrzeuge AG. 12
MES	Manufacturing Execution Systems. 49
MITI	Ministry of Trade and Industry. 23
Mitsubishi	Mitsubishi Motors Corporation. 23
MOQ	Minimum order quantity. 32, 39, 51
Nissan	Nissan Motor Co., Ltd. 23
PFEP	Plan for every part. 7, 21, 23–30, 35, 37, 46–51, 57, 59, 63, 65, 66, 69, 70
PLM	Product Lifecycle Management. 64
RWT	Reichweite. 7, 17–20, 32, 33, 35–39, 44, 51, 53, 63, 69

SaaS	Software as a Service. 64, 65
SCM	Supply chain management. 19, 21
SMED	Single Minute Exchange of Die. 23, 46
SQL	Structured Query Language. 28
Toyota	Toyota Motor Corporation. 7, 12, 23, 24
TPS	Toyota production system. 7, 23, 24, 46, 69
USD	Americký dolar. 64, 65
VBA	Visual Basic for Applications. 7, 40, 41, 69
Volvo	Volvo Cars. 12
VSM	Value stream mapping. 46
Witte	WITTE Automotive. 5–11, 13–19, 24, 26, 36, 37, 51, 66, 69
WMS	Warehouse management system. 49
ČR	Česká republika. 26, 77
Škoda	Škoda Auto. 12

Seznam příloh

Příloha A: VBA kód pro optimalizaci počtu kusů materiálu v L2 obale

Příloha B: Python kód pro simulaci RWT

Příloha C: Pohled na list s RWT simulací

Příloha A: VBA kód pro optimalizaci počtu kusů materiálu v L2 obale

```
1 Sub Obaly()  
2 Dim rg1 As Range  
3 Set rg1 = ActiveSheet.Range("F:N").CurrentRegion  
4  
5 Dim rg2 As Range  
6 Set rg2 = ActiveSheet.Range("P:S").CurrentRegion  
7  
8 Dim i As Long  
9 Dim arr(9) As Integer  
10 Dim arr1(3) As Integer  
11 Dim arr2(3) As Integer  
12 Dim arr3(3) As Integer  
13 For i = 2 To rg1.Rows.Count  
14  
15     'Definice globálního pole  
16     arr(0) = rg1.Cells(i, 1).Value2  
17     arr(1) = rg1.Cells(i, 2).Value2  
18     arr(2) = rg1.Cells(i, 3).Value2  
19     arr(3) = rg1.Cells(i, 4).Value2  
20     arr(4) = rg1.Cells(i, 5).Value2  
21     arr(5) = rg1.Cells(i, 6).Value2  
22     arr(6) = rg1.Cells(i, 7).Value2  
23     arr(7) = rg1.Cells(i, 8).Value2  
24     arr(8) = rg1.Cells(i, 9).Value2  
25  
26     'Definice pole1  
27     arr1(0) = rg1.Cells(i, 1).Value2  
28     arr1(1) = rg1.Cells(i, 2).Value2  
29     arr1(2) = rg1.Cells(i, 3).Value2  
30     'Definice pole2  
31     arr2(0) = rg1.Cells(i, 4).Value2  
32     arr2(1) = rg1.Cells(i, 5).Value2  
33     arr2(2) = rg1.Cells(i, 6).Value2  
34     'Definice pole3  
35     arr3(0) = rg1.Cells(i, 7).Value2  
36     arr3(1) = rg1.Cells(i, 8).Value2  
37     arr3(2) = rg1.Cells(i, 9).Value2  
38  
39     'Hledání maxima globálního pole  
40     max_val = WorksheetFunction.Max(arr)  
41     max_val_index = Application.Match(max_val, arr, False)  
42     switch_arr1 = arr1  
43     switch_arr2 = arr1  
44     switch_arr_str1 = 1  
45     switch_arr_str2 = 1  
46     'První interval má globální max  
47     If max_val_index <= 3 Then  
48         local_max1 = WorksheetFunction.Max(arr2)  
49         local_max2 = WorksheetFunction.Max(arr3)  
50         switch_arr1 = arr2  
51         switch_arr2 = arr3  
52         switch_arr_str1 = 2  
53         switch_arr_str2 = 3  
54  
55         If Application.Match(local_max1, arr2, False) = 1 And max_val_index = 1 Then  
56             local_max1 = WorksheetFunction.Large(arr2, 2)
```

```

57     ElseIf Application.Match(local_max1, arr2, False) = 2 And max_val_index = 2 Then
58         local_max1 = WorksheetFunction.Large(arr2, 2)
59     ElseIf Application.Match(local_max1, arr2, False) = 3 And max_val_index = 3 Then
60         local_max1 = WorksheetFunction.Large(arr2, 2)
61     End If
62
63     If Application.Match(local_max2, arr3, False) = 1 And max_val_index = 1 Then
64         local_max2 = WorksheetFunction.Large(arr3, 2)
65     ElseIf Application.Match(local_max2, arr3, False) = 2 And max_val_index = 2 Then
66         local_max2 = WorksheetFunction.Large(arr3, 2)
67     ElseIf Application.Match(local_max2, arr3, False) = 3 And max_val_index = 3 Then
68         local_max2 = WorksheetFunction.Large(arr3, 2)
69     End If
70 End If
71
72 'Druhý interval má globální max
73 If max_val_index > 3 And max_val_index <= 6 Then
74     local_max1 = WorksheetFunction.Max(arr1)
75     local_max2 = WorksheetFunction.Max(arr3)
76     switch_arr1 = arr1
77     switch_arr2 = arr3
78     switch_arr_str1 = 1
79     switch_arr_str2 = 3
80
81     If Application.Match(local_max1, arr1, False) = 1 And max_val_index = 4 Then
82         local_max1 = WorksheetFunction.Large(arr1, 2)
83     ElseIf Application.Match(local_max1, arr1, False) = 2 And max_val_index = 5 Then
84         local_max1 = WorksheetFunction.Large(arr1, 2)
85     ElseIf Application.Match(local_max1, arr1, False) = 3 And max_val_index = 6 Then
86         local_max1 = WorksheetFunction.Large(arr1, 2)
87     End If
88
89     If Application.Match(local_max2, arr3, False) = 1 And max_val_index = 4 Then
90         local_max2 = WorksheetFunction.Large(arr3, 2)
91     ElseIf Application.Match(local_max2, arr3, False) = 2 And max_val_index = 5 Then
92         local_max2 = WorksheetFunction.Large(arr3, 2)
93     ElseIf Application.Match(local_max2, arr3, False) = 3 And max_val_index = 6 Then
94         local_max2 = WorksheetFunction.Large(arr3, 2)
95     End If
96 End If
97
98 'Třetí interval má globální max
99 If max_val_index > 6 Then
100     local_max1 = WorksheetFunction.Max(arr1)
101     local_max2 = WorksheetFunction.Max(arr2)
102     switch_arr1 = arr1
103     switch_arr2 = arr2
104     switch_arr_str1 = 1
105     switch_arr_str2 = 2
106
107     If Application.Match(local_max1, arr1, False) = 1 And max_val_index = 7 Then
108         local_max1 = WorksheetFunction.Large(arr1, 2)
109     ElseIf Application.Match(local_max1, arr1, False) = 2 And max_val_index = 8 Then
110         local_max1 = WorksheetFunction.Large(arr1, 2)
111     ElseIf Application.Match(local_max1, arr1, False) = 3 And max_val_index = 9 Then
112         local_max1 = WorksheetFunction.Large(arr1, 2)
113     End If
114
115     If Application.Match(local_max2, arr2, False) = 1 And max_val_index = 7 Then
116         local_max2 = WorksheetFunction.Large(arr2, 2)
117     ElseIf Application.Match(local_max2, arr2, False) = 2 And max_val_index = 8 Then

```

```
118     local_max2 = WorksheetFunction.Large(arr2, 2)
119 ElseIf Application.Match(local_max2, arr2, False) = 3 And max_val_index = 9 Then
120     local_max2 = WorksheetFunction.Large(arr2, 2)
121 End If
122
123     End If
124
125 If (Application.Match(local_max1, switch_arr1, False) = Application.Match(local_max2,
    ↔ switch_arr2, False)) And local_max1 < local_max2 Then
126     local_max1 = WorksheetFunction.Large(switch_arr1, 3)
127 ElseIf (Application.Match(local_max1, switch_arr1, False) = Application.Match(local_max2,
    ↔ switch_arr2, False)) And local_max1 > local_max2 Then
128     local_max2 = WorksheetFunction.Large(switch_arr2, 3)
129
130 End If
131
132 rg2.Cells(i, 1).Value2 = WorksheetFunction.Max(arr)
133 rg2.Cells(i, 2).Value2 = local_max1
134 rg2.Cells(i, 3).Value2 = local_max2
135
136 Next i
137 End Sub
```

Příloha B: Python kód pro simulaci RWT

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sun Apr 26 18:24:44 2020
4
5 @author: richard.vanek
6 """
7 import time, random, csv, statistics, matplotlib.pyplot as plt, pyinputplus as pyip
8 from pulp import *
9 start_time = time.time()
10 print('Podpůrná simulace pro diplomovou práci "Studie logistického nástroje"\n Autor: Richard
    ↪ Vaněk\n')
11 automat = pyip.inputChoice(['1', '0'],prompt="Zadejte hodnotu 1 pro automatickou simulaci, hodnotu 0
    ↪ pro simulaci s uživatelským vstupem")
12 if automat == "1":
13     dolni_hrac = pyip.inputNum("Zadejte dolní hranici variability automatického hráče. Stiskněte
    ↪ ENTER pro výchozí hodnotu: 0.2", min=0.01, default=0.2, lessThan=1, limit=1)
14     horni_hrac = pyip.inputNum("Zadejte horní hranici variability automatického hráče. Stiskněte
    ↪ ENTER pro výchozí hodnotu: 0.8", min=0.01, default=0.8, lessThan=1, limit=1)
15 else:
16     pass
17 variabilita_pozadavku = pyip.inputNum("Zadejte variabilitu poptávky materiálu. Stiskněte ENTER pro
    ↪ výchozí hodnotu: 0.1", min=0.01, default=0.1, lessThan=1, limit=1)
18 pocet_obdobi = pyip.inputInt("Zadejte počet období. Stiskněte ENTER pro výchozí hodnotu: 12", min=1,
    ↪ default=12, limit=1)
19 variabilita_casu = pyip.inputNum("Zadejte variabilitu časů. Stiskněte ENTER pro výchozí hodnotu:
    ↪ 0.4", min=0.01, default=0.4, lessThan=1, limit=1)
20 skore_pocitace_var = []
21 skore_pocitace = []
22 skore_hrace = []
23 sum_hodnoty_zasob = []
24 sum_potreby = []
25 sum_pokuta = []
26 csv_export = []
27 RWT_list = []
28 LP_promenne = []
29
30 # Uživatelsky plánované denní průměrné skladové zásoby
31 prum_sklad_zas = {"Arm_insert" : 8500, "Arm_injected" : 10500, "Bracket" : 10800, "Pin_Bracket" :
    ↪ 12500, "Cover" : 7800, "Case" : 12600, "Bolt" : 35000}
32
33 for i in range(pocet_obdobi + 1):
34     # Definice random seedu
35     random.seed(a=random.random(), version=2)
36     # Definice hodnot pro LP úlohu
37     goal_RWT = random.randint(4, 6)
38     RWT_list.append(goal_RWT)
39     ceny_za_dil = [1.53, 2.12, 3.40, 1.10, 4.20, 5.10, 0.70]
40     d_prum_hod_poz = [2142, 2968, 4760, 1540, 5880, 7140, 1960]
41     cas_na_dodani_materialu = [6, 7, 7, 5, 5, 7, 7]
42
43     # Definice proměnlivosti výrobní poptávky po materiálu
44     for y in range(len(d_prum_hod_poz)):
45         d_prum_hod_poz[y] = int(round((d_prum_hod_poz[y] +
    ↪ d_prum_hod_poz[y]*random.uniform(variabilita_pozadavku*-1, variabilita_pozadavku), 0))
46     # Externí šok na dodací dobu materiálu
47     pocet_soku_cas = (random.choices([1, 2, 3], weights=[7, 3, 2], k=1))
48     for w in range(0, pocet_soku_cas[0]):
```

```

49     cas_na_dodani_materialu[random.randrange(0, len(cas_na_dodani_materialu))] =
        ↪ round((cas_na_dodani_materialu[random.randrange(0, len(cas_na_dodani_materialu))] +
        ↪ cas_na_dodani_materialu[random.randrange(0,
        ↪ len(cas_na_dodani_materialu))] * random.uniform(variabilita_casu*-1, variabilita_casu)),
        ↪ 0)
50     # Model LP
51     # Definice Max/Min
52     prob = LpProblem("RWT", LpMinimize)
53
54     # Definice proměnných
55     sz1 = LpVariable("sz1", lowBound=0, cat='Integer')
56     sz2 = LpVariable("sz2", lowBound=0, cat='Integer')
57     sz3 = LpVariable("sz3", lowBound=0, cat='Integer')
58     sz4 = LpVariable("sz4", lowBound=0, cat='Integer')
59     sz5 = LpVariable("sz5", lowBound=0, cat='Integer')
60     sz6 = LpVariable("sz6", lowBound=0, cat='Integer')
61     sz7 = LpVariable("sz7", lowBound=0, cat='Integer')
62
63
64     # Definice účelové funkce
65     prob += ceny_za_dil[0]*sz1 + ceny_za_dil[1]*sz2 + ceny_za_dil[2]*sz3 + ceny_za_dil[3]*sz4 +
        ↪ ceny_za_dil[4]*sz5 + ceny_za_dil[5]*sz6 + ceny_za_dil[6]*sz7
66     # Omezující podmínky
67     prob += ((sz1*ceny_za_dil[0])/d_prum_hod_poz[0]) >= cas_na_dodani_materialu[0]
68     prob += ((sz2*ceny_za_dil[1])/d_prum_hod_poz[1]) >= cas_na_dodani_materialu[1]
69     prob += ((sz3*ceny_za_dil[2])/d_prum_hod_poz[2]) >= cas_na_dodani_materialu[2]
70     prob += ((sz4*ceny_za_dil[3])/d_prum_hod_poz[3]) >= cas_na_dodani_materialu[3]
71     prob += ((sz5*ceny_za_dil[4])/d_prum_hod_poz[4]) >= cas_na_dodani_materialu[4]
72     prob += ((sz6*ceny_za_dil[5])/d_prum_hod_poz[5]) >= cas_na_dodani_materialu[5]
73     prob += ((sz7*ceny_za_dil[6])/d_prum_hod_poz[6]) >= cas_na_dodani_materialu[6]
74     prob += ((ceny_za_dil[0]*sz1 + ceny_za_dil[1]*sz2 + ceny_za_dil[2]*sz3 + ceny_za_dil[3]*sz4 +
        ↪ ceny_za_dil[4]*sz5 + ceny_za_dil[5]*sz6 + ceny_za_dil[6]*sz7) / sum(d_prum_hod_poz)) >=
        ↪ goal_RWT
75
76     status = prob.solve()
77     print(LpStatus[status])
78     skore_pocitace.append(int(round(value(prob.objective),0)))
79     externi_sok_zas = random.choices([random.uniform(0.001, 0.1), random.uniform(0.001, 0.2)],
        ↪ weights=[10, 3], k=1)
80     skore_pocitace_var.append(int(round((value(prob.objective) +
        ↪ value(prob.objective)*externi_sok_zas[0]),0)))
81     print(value(prob.objective)/(sum(d_prum_hod_poz)))
82
83     # Hodnoty proměnných
84     for e in range(7):
85         sz_name = "sz" + str(e + 1)
86         LP_promenne.append(value(eval(sz_name)))
87
88
89     # Definice logiky hráče
90     print("Prosím vyplňte průměrnou plánovanou denní skladovou zásobu u každé položky materiálu.")
91     print("Na vyplnění každé hodnoty máte 15s, po uplynutí limitu se nastaví výchozí hodnota.")
92     if automat == "0":
93         input("Pro pokračování stiskněte ENTER")
94     hodnoty_zasob = []
95     potreby = []
96     pokuta = []
97     counter_list = -1
98     for x in prum_sklad_zas:
99         counter_while = 0
100        counter_list += 1

```

```

101     while counter_while < 3:
102         print("Cílové globální RWT: " + str(goal_RWT))
103         print("Aktuální globální RWT: " + str(sum(prum_sklad_zas.values())/sum(d_prum_hod_poz)))
104         print("Aktuální RWT materiálu: " + str(prum_sklad_zas[x]/d_prum_hod_poz[counter_list]))
105         print("Dodací doba materiálu: " + str(cas_na_dodani_materialu[counter_list]) + " d")
106         print("Jednotkové náklady na materiál: " + str(ceny_za_dil[counter_list]) + " €")
107         print("Hodnota dodávky: " + str(prum_sklad_zas[x]*ceny_za_dil[counter_list]) + " €")
108         print("\nPočet kusů materiálu - " + x + " : " + str(prum_sklad_zas[x]))
109         print("Materiál: " + str(counter_list + 1) + "/" + str(len(cas_na_dodani_materialu)))
110         print("Období: " + str(i + 1) + "/" + str(pocet_obdobi))
111         if automat == "1":
112             #response = 0
113             response = random.randrange(LP_promenne[counter_list] -
114                                     ↪ int(round(dolni_hrac*LP_promenne[counter_list], 0)), LP_promenne[counter_list]
115                                     ↪ + int(round(horni_hrac*LP_promenne[counter_list], 0)))
116         else:
117             response = pyip.inputInt(timeout=15, min=0, default=prum_sklad_zas[x])
118             prum_sklad_zas[x] = response
119             print("Zadáno: " + str(response))
120             if automat == "0":
121                 input("Pokračujte stisknutím ENTER")
122             counter_while += 1
123             hodnoty_zasob.append(prum_sklad_zas[x]*ceny_za_dil[counter_list])
124             potreby.append(d_prum_hod_poz[counter_list]*cas_na_dodani_materialu[counter_list])
125             if (prum_sklad_zas[x]) < LP_promenne[counter_list]:
126                 pokuta.append(((cas_na_dodani_materialu[counter_list]*d_prum_hod_poz[counter_list]) -
127                             ↪ prum_sklad_zas[x])*ceny_za_dil[counter_list]*0.5)
128             else:
129                 pokuta.append(0)
130             # Pokuta
131             if sum(hodnoty_zasob) + sum(pokuta) < value(prob.objective):
132                 skore_hrace.append(int(round((value(prob.objective) + sum(pokuta)), 0)))
133             else:
134                 skore_hrace.append(int(round((sum(hodnoty_zasob) + sum(pokuta)), 0)))
135             sum_hodnoty_zasob.append(sum(hodnoty_zasob))
136             sum_potreby.append(sum(potreby))
137             sum_pokuta.append(sum(pokuta))
138
139     print("--- %s seconds ---" % (time.time() - start_time))
140     print("Skóre hráče: " + '{:,}'.format(sum(skore_hrace)).replace(',',' '))
141     print("Skóre počítače - var: " + '{:,}'.format(sum(skore_pocitace_var)).replace(',',' '))
142     print("Skóre počítače: " + '{:,}'.format(sum(skore_pocitace)).replace(',',' '))
143     print("Rozdíl: " + '{:,}'.format(sum(skore_hrace) - sum(skore_pocitace)).replace(',',' '))
144     print("Medián hráče: " + '{:,}'.format(statistics.median(skore_hrace)).replace(',',' '))
145     print("Medián počítače - var: " + '{:,}'.format(statistics.median(skore_pocitace_var)).replace(',',' '
146           ↪ '))
147     print("Medián počítače: " + '{:,}'.format(statistics.median(skore_pocitace)).replace(',',' '))
148     # line 1 points
149     x1 = [z for z in range(len(skore_pocitace_var))]
150     y1 = skore_pocitace_var
151     # plotting the line 1 points
152     plt.plot(x1, y1, label = "Skóre počítače - variabilita")
153
154     # line 2 points
155     x2 = [z for z in range(len(skore_pocitace))]
156     y2 = skore_pocitace
157     # plotting the line 2 points
158     plt.plot(x2, y2, label = "Skóre počítače - ideál")
159
160     # line 3 points
161     x3 = [z for z in range(len(skore_hrace))]

```



```

158 y3 = skore_hrace
159 # plotting the line 2 points
160 plt.plot(x3, y3, label = "Skóre hráče")
161
162 # line 4 points
163 x4 = [z for z in range(len(sum_pokuta))]
164 y4 = sum_pokuta
165 # plotting the line 2 points
166 plt.plot(x4, y4, label = "Pokuta")
167 """
168 # line 5 points
169 x5 = [z for z in range(len(sum_hodnoty_zasob))]
170 y5 = sum_hodnoty_zasob
171 # plotting the line 2 points
172 plt.plot(x5, y5, label = "Hodnoty zásob")
173
174 # line 5 points
175 x6 = [z for z in range(len(sum_potreby))]
176 y6 = sum_potreby
177 # plotting the line 2 points
178 plt.plot(x6, y6, label = "Potřeby")
179 """
180 # naming the x axis
181 plt.xlabel('Počet období')
182 # naming the y axis
183 plt.ylabel('Hodnota v €')
184 # giving a title to my graph
185 plt.title('Výsledná hodnota zásob')
186
187 # show a legend on the plot
188 plt.legend()
189
190 # function to show the plot
191 plt.savefig('chart.png', bbox_inches="tight", dpi=300)
192 plt.show()
193
194 # Příprava seznamu pro export do CSV
195 for i in range(len(skore_hrace)):
196     if i == 0:
197         csv_export.append(["Skore_hrace", "Skore_pocitace_var", "Skore_pocitace"])
198     else:
199         csv_export.append([skore_hrace[i], skore_pocitace_var[i], skore_pocitace[i]])
200
201 with open("skore.csv", "w") as data_file:
202     csv_writer = csv.writer(data_file, delimiter=",")
203     for line in csv_export:
204         csv_writer.writerow(line)

```

Příloha C: Pohled na list s RWT simulací

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1																									
2																									
3	Look up_d ata	Disponent.zk ata	Disponent	Znak analyzy ABC	Číslo dílu	Název dílu	Skladová zásoba (ks)	Cena za 1ks dílu (EUR)	Hodnota skladové zásoby (EUR)	Denní průměrná hodnota požadavků (EUR)	RWT	Díleční stanovené cílové RWT	Hodnota plánované úspory	Plánovaná denní průměrná hodnota skladové zásoby (ks)	RWT + poj. Zás. Delta	Díleční stanovené cílové RWT + poj. Zás.	Hodnota plánované úspory + poj. Zás.	Plánovaná denní průměrná hodnota skladové zásoby + poj. Zás. (ks)			Pojistná zásoba (ks)	Čas na dodání materiálu (dny)			
4	3	NIA	Novák, Jan	A	01056101714	Arm_insert	10 919,00	1,53	16 706,07	2 142,00	7,80	6,00	3 854,07	12 852,00	2,50	8,50 - 1 500,93	8,50 - 1 500,93	18 207,00	11 900,00	3 500,00	6				
5	4	NJA	Novák, Jan	A	01165623940	Arm_injected	11 359,00	2,12	24 081,08	2 968,00	8,11	7,00	3 305,08	20 776,00	2,86	9,86 - 5 174,92	9,86 - 5 174,92	29 256,00	13 800,00	4 000,00	7				
6	5	NJA	Novák, Jan	A	01086752104	Bracket	13 670,00	3,40	46 478,00	4 760,00	9,76	5,00	13 158,00	33 320,00	1,67	8,67 - 5 225,80	8,67 - 5 225,80	41 252,20	12 133,00	2 333,00	7				
7	6	NJA	Novák, Jan	A	01194770930	Pin_Bracket	13 890,00	1,10	15 279,00	1 540,00	9,92	5,00	7 579,90	7 700,00	-2,14	7,14 - 4 279,00	7,14 - 4 279,00	11 000,00	10 000,00	3 000,00	5				
8	7	NJA	Novák, Jan	A	01192210641	Cover	13 197,00	4,20	55 427,40	5 880,00	9,43	5,00	26 027,40	29 400,00	7,00	6,79 - 15 527,40	6,79 - 15 527,40	39 900,00	9 500,00	2 500,00	6				
9	8	NJA	Novák, Jan	A	01049604346	Case	10 944,00	5,10	55 814,40	7 140,00	7,82	7,00	5 834,40	49 980,00	1,79	8,07 - 1 815,60	8,07 - 1 815,60	57 630,00	11 300,00	1 500,00	9				
10	9	NJA	Novák, Jan	A	01031732053	Bolt	22 438,00	0,70	15 706,60	1 960,00	8,01	7,00	1 986,60	13 720,00	1,79	8,79 - 1 513,40	8,79 - 1 513,40	17 220,00	24 600,00	5 000,00	12				
11																									
12																									
13																									
14																									
15																									
16																									
17																									
18																									
19																									
20																									
21																									
22																									
23																									
24																									
25																									
26																									
27																									
28																									
29																									
30																									
31																									
32																									
33																									
34																									
35																									
36																									
37																									

Zdroj: Vlastní zpracování

Absrakt

VANĚK, R. *Studie implementace logistického nástroje*. Diplomová práce. Plzeň: Fakulta ekonomická ZČU v Plzni, s. 82, 2020

Klíčová slova

ZČU, FEK, PFEP, Logistika, Plan for every part

Abstrakt

Tato diplomová práce je zaměřena na jednu z méně probádaných částí Toyota production system (TPS), kterou je nástroj Plan for every part (PFEP). První část práce je věnována představení spolupracujícího subjektu, jeho historie a oblasti podnikání. Druhá část zasazuje PFEP do kontextu historie TPS a popisuje navrhovanou strukturu nástroje. Třetí část je věnována postupu implementace nástroje PFEP. Poslední část práce se poté věnuje simulaci zavedení nástroje a ekonomickému zhodnocení.

Abstract

VANĚK, R. *Logistics tool implementation study*. Master thesis. Pilsen: Faculty of Economics, University of West Bohemia, s. 82, 2020

Keywords

WB, FEK, PFEP, Logistics, Plan for every part

Abstract

This master's thesis is focused on one of the less explored parts of the Toyota production system (TPS), specifically the Plan for every part (PFEP). The first part is devoted to the introduction of the cooperating firm, its history and its field of business. The second part puts PFEP in the context of TPS history and describes the proposed structure of the tool. The third part is devoted to the process of implementation of the PFEP tool. The last part of the work is then focused on the simulation which deals with the effects of implementation of the tool and lastly a final economic evaluation is performed.