



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

VYHODNOCENÍ A VIZUALIZACE VÝROBNÍCH TECHNICKÝCH ZTRÁT VYČÍTANÝCH Z DATABÁZE MES

EVALUATION AND VISUALIZATION OF PRODUCTION TECHNICAL LOSSES READ FROM THE MES
DATABASE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Jiří Šlejtr

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Mgr. Veronika Novotná, Ph.D.

BRNO 2025

Zadání bakalářské práce

Ústav: Ústav informatiky
Student: Jiří Šlejtr
Vedoucí práce: doc. Mgr. Veronika Novotná, Ph.D.
Akademický rok: 2024/25
Studijní program: Manažerská informatika

Garant studijního programu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává bakalářskou práci s názvem:

Vyhodnocení a vizualizace výrobních technických ztrát vyčítaných z databáze MES

Charakteristika problematiky úkolu:

Úvod
Cíle práce, metody a postupy zpracování
Teoretická východiska práce
Analýza současného stavu
Vlastní návrhy řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Cílem práce je vytvoření nové vizualizace výrobních ztrát v Power BI na základě dat z databáze OISNet (MES). Data se budou zobrazovat v reálném čase. Datová základna musí být snadno rozšiřitelná na další příchozí výroby.

Základní literární prameny:

LACKO, Luboslav. Databáze: datové sklady, OLAP a dolování dat s příklady v Microsoft SQL Serveru a Oracle. Brno: Computer Press, 2003. ISBN 8072269690.

NOVOTNÝ, Ota, Jan POUR a David SLÁNSKÝ. Business intelligence: jak využít bohatství ve vašich datech. Praha: Grada, 2005. ISBN 80-247-1094-3.

POUR, Jan, Miloš MARYŠKA a Ota NOVOTNÝ. Business intelligence v podnikové praxi. Praha: Professional Publishing, 2012. ISBN 978-80-7431-065-2.

POUR, Jan, Miloš MARYŠKA, Iva STANOVSKÁ a Zuzana ŠEDIVÁ. Self service business intelligence: jak si vytvořit vlastní analytické, plánovací a reportingové aplikace. Grada Publishing, 2018. ISBN ISBN 978-80-271-0616-5.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2024/25

V Brně dne 9.2.2025

L. S.

Ing. Jiří Kříž, Ph.D.
garant

prof. Ing. et Ing. Stanislav Škapa, Ph.D.
děkan

ABSTRAKT

Tato bakalářská práce se zaměřuje na využití analytických a reportovacích služeb ve firmách. V práci je nejprve provedena analýza firemního prostředí a následně je využita k tvorbě reportu o výkonových ztrátách v aplikaci Power BI pro firmu Bosch Powertrain s.r.o., který následně bude sloužit pro efektivnější a informovanější manažerské rozhodování ve firmě.

KLÍČOVÁ SLOVA

Microsoft Power BI, Business Intelligence, report, data, informace, vizualizace

ABSTRACT

This bachelor thesis focuses on the use of analytics and reporting services in companies. The thesis first analyses the corporate environment and then uses it to create a report on performance losses in Power BI for Bosch Powertrain s.r.o., which will then be used to make more effective and informed managerial decisions in the company.

KEYWORDS

Microsoft Power BI, Business Intelligence, report, data, information, visualization

BIBLIOGRAFICKÁ CITACE

ŠLEJTR, Jiří. *Vyhodnocení a vizualizace výrobních technických ztrát vyčítaných z databáze MES*. Online, bakalářská práce. Veronika NOVOTNÁ (vedoucí práce). Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2025. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/168604>. [cit. 2025-05-01].

PROHLÁŠENÍ

Prohlašuji, že předložená bakalářská práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

.....

Datum

.....

Jméno a příjmení

PODĚKOVÁNÍ

Rád bych touto cestou poděkoval svému školiteli z firmy panu Mgr. Patriku Bakusovi za jeho cenné rady při zpracování této bakalářské práce a firmě za příležitost zpracovat tuto bakalářskou práci. Dále bych chtěl poděkovat své přítelkyni za podporu během celého studia.

OBSAH

ÚVOD.....	11
CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ.....	12
1. TEORETICKÁ VÝCHODISKA PRÁCE.....	13
1.1. Business Intelligence	13
1.1.1. Historie a vývoj BI.....	13
1.1.2. Architektura BI	14
1.2. Data, informace, znalosti	15
1.2.1. Data	15
1.2.2. Informace	15
1.2.3. Znalost	16
1.3. ETL (Extraction, Transformation, Loading).....	16
1.3.1. Extrakce	16
1.3.2. Transformace	18
1.3.3. Přenos dat (Loading).....	19
1.4. Datový sklad	19
1.5. Datové tržiště	20
1.6. Databáze.....	20
1.6.1. Relační databáze	21
1.6.2. Jazyk SQL.....	21
1.7. OLAP.....	22
1.7.1. OLAP kostka.....	23
1.7.2. Dimenze a fakta	23
1.8. Schémata	24
1.8.1. Schéma hvězdy	24

1.8.2.	Schéma sněhové vločky	25
1.9.	Microsoft Power BI	25
1.9.1.	Komponenty Power BI	25
1.9.2.	Základní pojmy Power BI.....	27
1.9.3.	Data Analysis Expressions (DAX)	28
1.10.	Metody analýzy.....	29
1.10.1.	SWOT analýza.....	29
1.10.2.	Paretova analýza	29
1.10.3.	PDCA cyklus	30
2.	ANALÝZA SOUČASNÉHO STAVU	32
2.1.	Historie společnosti.....	32
2.2.	Vlastnická struktura	33
2.3.	Popis jihlavské společnosti	34
2.4.	Analýza vnitřního prostředí firmy	35
2.5.	Popis současného stavu IS	36
2.6.	Analýza využití komponent BI	39
2.6.1.	SWOT analýza reportovacích a analytických služeb	39
3.	VLASTNÍ NÁVRH ŘEŠENÍ	42
3.1.	Požadavky	42
3.2.	Datové zdroje.....	42
3.3.	Načtení dat do Power BI.....	46
3.4.	Transformace dat pro vizualizaci v Power BI	51
3.4.1.	Tabulky s výkonovými ztrátami	52
3.4.2.	Tabulky s denním OEE.....	53
3.4.3.	Tabulky s měsíčním OEE	54

3.4.4.	Tabulka CalendarYear	56
3.4.5.	Tabulka KPI_Opatření	57
3.5.	Vytvoření schéma datového modelu.....	57
3.6.	Vytvořené míry a vizualizace	58
3.6.1.	Top 10 Ztrát report.....	59
3.6.2.	Report PDCA SMMC	60
3.6.3.	Report OEE Overview	61
3.6.4.	Report denní sledování	62
	ZÁVĚR	63
	SEZNAM POUŽITÉ LITERATURY	64
	SEZNAM OBRÁZKŮ.....	67
	SEZNAM TABULEK	69

ÚVOD

V současné době hrají informační technologie velkou roli v životě každého z nás, aniž bychom si to uvědomovali. Informační technologie využíváme od samotného rána, kdy vezmeme do ruky mobilní telefon. S informačními technologiemi se velká část populace setkává i ve svém pracovním životě. Informační technologie nám mají zjednodušit pracovní činnost. Díky využívání obrovského množství informačních technologií získáváme velké množství dat. Vždy se musíme rozhodnout, jak dále s daty naložíme, mohou nám dát nepřehledné množství možností, které mohou vést k zefektivnění činnosti či zvýšení výdělků, anebo nám nemusí dát vůbec nic a mohou nám připadat zbytečná a zatěžující. Díky správné interpretaci se data proměňují na informace a na znalosti, které dále využíváme pro náš pracovní život.

Datové analýzy a správné zacházení s daty může podpořit a zlepšit manažerské rozhodování ve firmách, jelikož se rozhodnutí přijímají na základě analýz a dat, a ne na základě subjektivních pocitů a odhadů. Každý den vzniká nepřehledné množství dat, které není člověk sám schopný zanalyzovat pouhým pohledem na ně. Proto se využívá umělá inteligence a další analytické nástroje, které dokáží zpracovat velké množství dat a vytvořit z nich výstupy.

Ve firmách je k analyzování dat často využívaná technologie Business Intelligence, která nám umožňuje vytvořit z dat informace a z nich znalosti. Business Intelligence ve formě reportů a analýz je využívána ve firmách k efektivnímu a informovanému rozhodování.

V této bakalářské práci se zaměřím na popsání pojmu Business Intelligence a pojmy s ním související, zanalyzuji výchozí stav ve firmě Bosch Powertrain s.r.o. a navrhnou řešení pro efektivnější analýzu dat vedoucí k lepšímu rozhodování.

CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ

Cílem práce je vytvoření nové vizualizace výrobních ztrát v Power BI na základě dat z databáze OISNet (MES). Data se budou zobrazovat v reálném čase. Datová základna musí být snadno rozšiřitelná na další příchozí výroby.

Práce se skládá ze tří částí. První část obsahuje teoretická východiska, kde je například popsán vznik pojmu Business Intelligence a jeho historie. Dále je v ní zmíněno, jak z dat vznikají informace a z nich následně znalosti. Zmíněn je tu také proces ETL a databázové systémy, bez kterých by Business Intelligence nemohlo fungovat. V další části teoretických východisek je představena aplikace Power BI a pojmy s ní související. Na závěr jsou představeny metody analýz, které byly využité v dalších částech práce.

Druhá část je nazvána analýza současného stavu, popisují v ní společnost Bosch a její historii, dceřinou společnost Bosch Powertrain s.r.o., ve které pracuji a podílím se na tvorbě reportů. Zanalyzováno je zde vnitřní prostředí firmy, současný stav využívání IS a Business Intelligence. Poslední částí je analýza SWOT, ze které vychází návrh řešení.

Poslední částí bakalářské práce je vlastní návrh řešení, který je zpracován v aplikaci Power BI. Zde se lze dozvědět proces získávání a transformování dat, vytváření schematického modelu a vizuálů.

Výsledný návrh řešení by měl být využíván k efektivnímu a informovanému rozhodování vedoucích pracovníků ve firmě.

1. TEORETICKÁ VÝCHODISKA PRÁCE

Teoretická část této bakalářské práce se zabývá základními pojmy, zmíněnými v cílech práce, které jsou důležité pro pochopení celé práce.

1.1. Business Intelligence

Business intelligence (BI) je termín, pod který spadá celá řada činností, úloh a technologií, které jsou dnes velmi často využívány pro podporu lepších rozhodovacích možností a jsou součástí řízení podniku a informačních systémů v podnicích. Hlavním cílem BI je zlepšit a zjednodušit přístup k datům i v reálném čase a umožnit podnikovým analytikům a vedoucím pracovníkům s nimi pracovat. [1, s. 13]

Rozhodovací orgány získávají díky analýze dat užitečné znalosti, které jim pomáhají dělat informované a efektivní rozhodnutí. V silné konkurenci musí manažeři rozhodovat rychle a zodpovědně, proto potřebují znalosti založené na aktuálních a spolehlivých informacích v reálném čase. [1, s. 13-14]

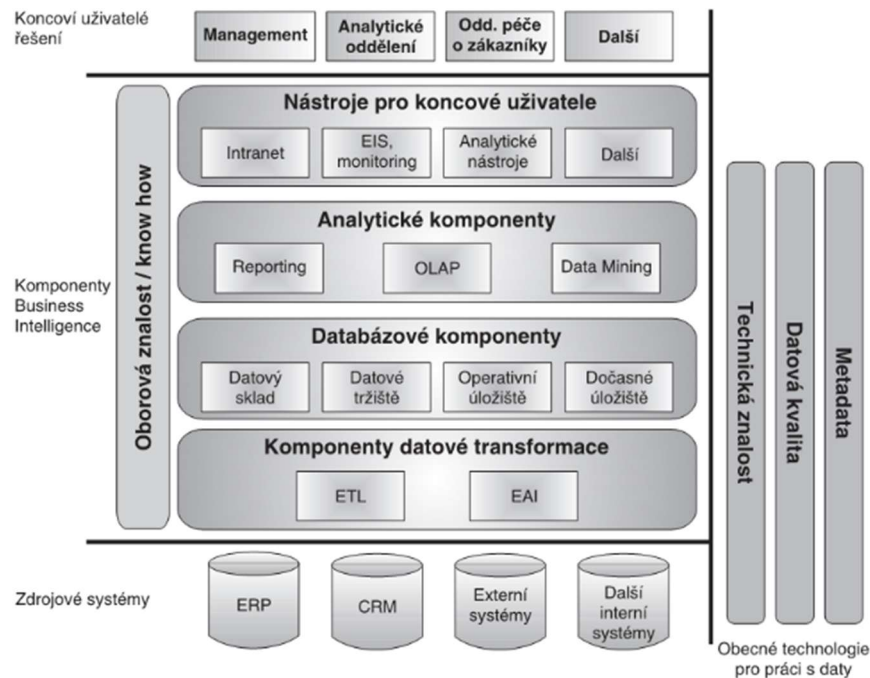
1.1.1. Historie a vývoj BI

První využití podobné dnešnímu BI se poprvé objevily na konci sedmdesátých let minulého století. První pokusy zapojit manažerské a analytické úlohy v podnikovém řízení jsou spojeny s firmou Lockheed. V osmdesátých letech došlo k velkému rozmachu BI, byly publikovány první významné práce o BI. V druhé polovině osmdesátých let přišly firmy Comshare a Pilot s prvními komerčními produkty, založenými na uložení a zpracování dat, známými pod názvem EIS (Executive Information System). V České republice se první systémy začaly objevovat okolo roku 1993. [1, s. 17]

Během přelomu osmdesátých a devadesátých let přišly na trh datové sklady a tržiště, což znamenalo rozvoj dalšího trendu v multidimenzionálních technologiích. Na českém trhu se tyto technologie začaly prosazovat až v druhé polovině devadesátých let. S velkým nárůstem objemu dat v datových skladech se začaly prosazovat techniky tzv. dolování dat založené na vysoce propracovaných analýzách dat za pomoci matematických a statistických metod. [1, s. 17]

1.1.2. Architektura BI

Od začátku vývoje BI se ustálila obecná koncepce architektury. Lze říct, že každý systém obsahuje několik určitých vrstev (Obrázek č. 1), které se dají podle potřeb podniku přizpůsobovat. Tyto části BI se neustále vyvíjí v průběhu let na základě uživatelských recenzí. [1, s. 26]



Obrázek č. 1: Obecná koncepce architektury BI

(Zdroj: [1, s. 27])

- Vrstva pro extrakci, transformaci a čištění dat (komponenty datové transformace) pokrývají oblasti sběru/přenosu dat ze zdrojových systémů do vrstvy pro ukládání dat. Konkrétní příklady jsou ETL (Extraction, Transformation, Loading) a EAI (Enterprise Application Integration).
- Vrstva pro ukládání dat obsahuje databázové komponenty (datové sklady, datová tržiště atd.), které se starají o ukládání a aktualizaci dat.
- Vrstva pro analýzu dat obsahuje analytické komponenty (reporting, OLAP analýza, data mining).
- Prezentační vrstva (nástroje pro koncové uživatele) zajišťuje komunikaci mezi koncovými uživateli a ostatními komponentami BI -> sběr informací pro analytické operace a následná prezentace výsledků.

- Vrstva oborové znalosti (oborová znalost, know-how) nejlepší nasazování BI pro konkrétní situaci ve firmě. [1, s. 26-27]

1.2. Data, informace, znalosti

Data, informace, znalosti jsou dnes spojeny s naším každodenním životem a každý člověk si pod těmito pojmy může představit něco jiného. Tyto údaje bývají lehce zaměnitelné s jinými pojmy, jako jsou například novinka, údaj atd. Proto jsou následující kapitoly věnovány vysvětlení těchto pojmů.

1.2.1. Data

Data lze popsat jako objektivní fakta o událostech. Mohou to být čísla, písmena, symboly atd. Data se dají získat experimentováním, měřením, pozorováním anebo šetřením. K datům se snažíme přiřadit význam a interpretovat je. Data se dají využít jako jednoduchá reprezentace faktů s jedinečným významem. [2]

Data dělíme na strukturovaná a nestrukturovaná:

- Strukturovaná data – zachycují fakta, atributy a objekty. Tato data se poté ukládají do relačních databázových systémů, ve kterých se poté využívá hierarchie elementů pole, záznam, relace, databáze. Díky organizaci dat lze poté vybrat jen relevantní data.
- Nestrukturovaná data – tok bitů bez další specifikace jako je například: videozáznam, zvuková nahrávka, obrázek nebo textový dokument. [2]

1.2.2. Informace

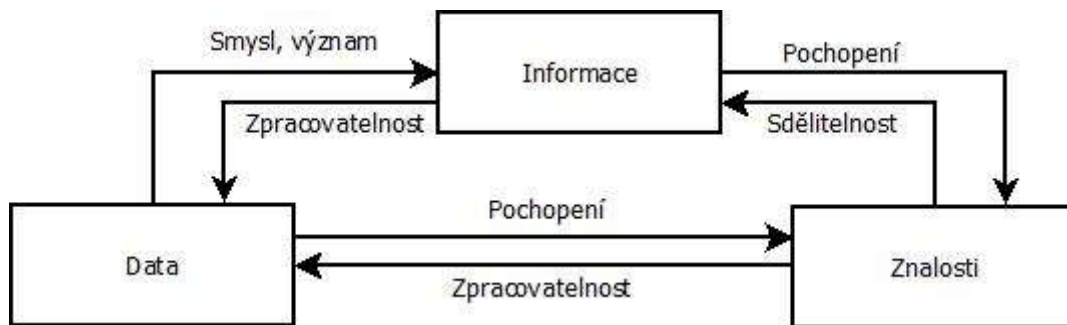
Informace prostupují všemi oblastmi lidských činností, z čehož vyplývá že pojem informace je složité definovat. Pojem informace pochází z latinského „informatio“, které pochází ze slova „informare“, znamenající informovat. Ve středověkém překladu tento termín znamenal dát formu myšlenky. Již od samého vzniku informace popisuje určité zhmotnění myšlenky do interpretovatelné podoby. Další pohled na informace je, že informace jsou data, které příjemce na základě svých znalostí a zkušeností, dokáže interpretovat s významem. Pokud výraz informace systematizujeme můžeme definovat hmotnou a nehmotnou informaci. Když se můžeme informace dotknout, není to znalost, ale hmotná informace například papírový dokument. V dnešní elektronické době

a elektronického zpracování informací, může informace představovat informační systém, kde jsou informace zpracovávány. [2]

1.2.3. Znalost

Znalost neboli poznání, vědění vyplývá z porozumění zákonitostem. Pokud je informace uspořádána v lidské mysli a s přidáním hodnotou, tak aby se dala využívat, hovoříme o znalosti. Znalosti jsou založené na interpretaci, zkušenostech a také záleží na samotném člověku, jak je schopný si dávat informace do souvislostí. Znalosti získáváme ze školy, ze svých vlastních zkušeností a obtížně se předávají. [2]

Pro lepší představivost o vzájemném propojení mezi těmito pojmy přikládám obrázek (Obrázek č. 2). Data jsou symboly představující výrobní surovinu, jež přechází do informace a znalosti poté interpretují data a informace. [2]



Obrázek č. 2: Vztahy mezi daty, informacemi a znalostmi

(Zdroj: Vlastní zpracování dle [2])

1.3. ETL (Extraction, Transformation, Loading)

Proces ETL je jednou z nejdůležitějších částí BI, známý také jako datová pumpa. Úkolem procesu ETL je vybrat data ze zdrojových systémů (Extraction), poté data upravit do požadovaného tvaru (Transformation) a poté je nahrát do specifikovaných datových úložišť (Loading). Nástroje ETL lze použít pro přenos mezi dvěma či více libovolnými databázemi, tyto systémy většinou pracují na principu dávkovacích režimů, data jsou přenášena v jedné chvíli buď denně, týdně nebo měsíčně. [1; 3]

1.3.1. Extrakce

Extrakce dat se provádí z archivních dat nebo z nehomogenního operačního prostředí. Nejprve se extrakce dělá z archivních dat, poté buď periodicky z interních dat společnosti nebo nárazově z externích dat (např. z internetu). [4]

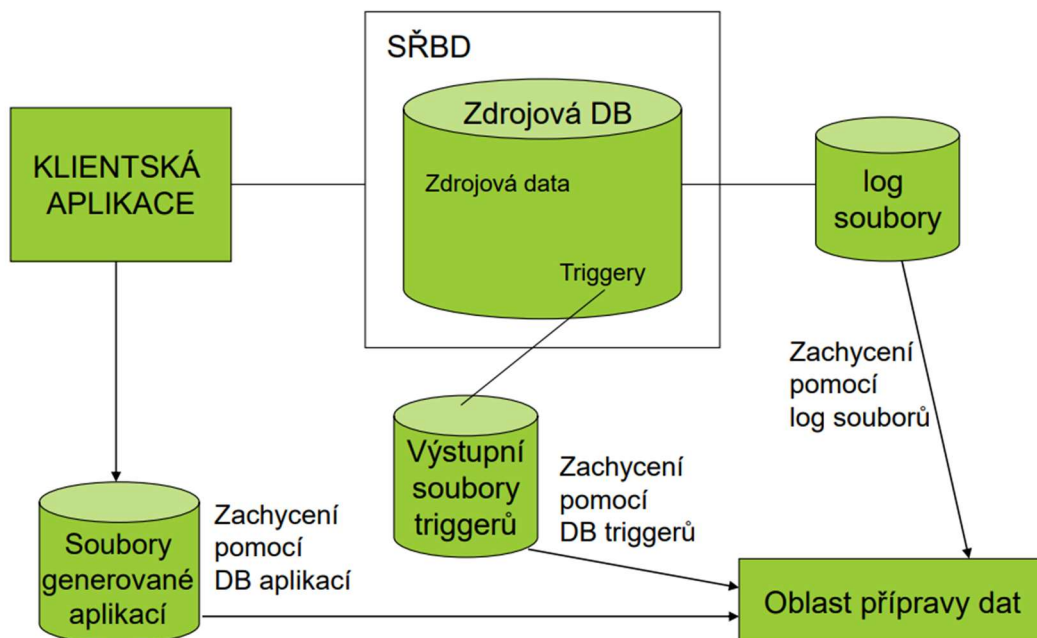
Hlavními cíli procesu extrakce jsou identifikace struktury zdrojů, stanovení metody extrakce pro každý zdroj, buď ručně (pomocí SQL) anebo pomocí nástroje, frekvence extrakcí a stanovení časového okna, kdy má extrakce probíhat a jak má probíhat, jestli sériově nebo paralelně. [5]

1.3.1.1. Metody extrakce

Extrakce může probíhat několika způsoby, základními metodami jsou extrakce statických dat a extrakce při aktualizaci dat. Metody extrakce statických dat spočívají ve vytvoření obrazu zdrojové databáze na výstupu a používá se při iniciálním nahrávání do databáze/skladu. Při využívání metod extrakce při aktualizaci dat je potřeba rozlišit nově vložená data od starších již extrahovaných. Tyto metody se dají rozdělit na metody přímé a odložené extrakce. [4]

1.3.1.1.1. Metody přímé extrakce

Metody přímé extrakce zachycují provedení změn ve zdrojových datech ihned (Obrázek č. 3). Pro zachycení provedených změn v databázích slouží různé způsoby. Jedním z nich je zachycení změn pomocí log souborů, které jsou vytvořené samotnou databází. Dalším způsobem je zachycení pomocí databázových triggerů respektive posloupnosti SQL příkazů, které zaznamenají změnu a zapíšíou ji do souboru. Posledním způsobem je možné zachytit změny v databázi pomocí samostatných databázových aplikací. [4]

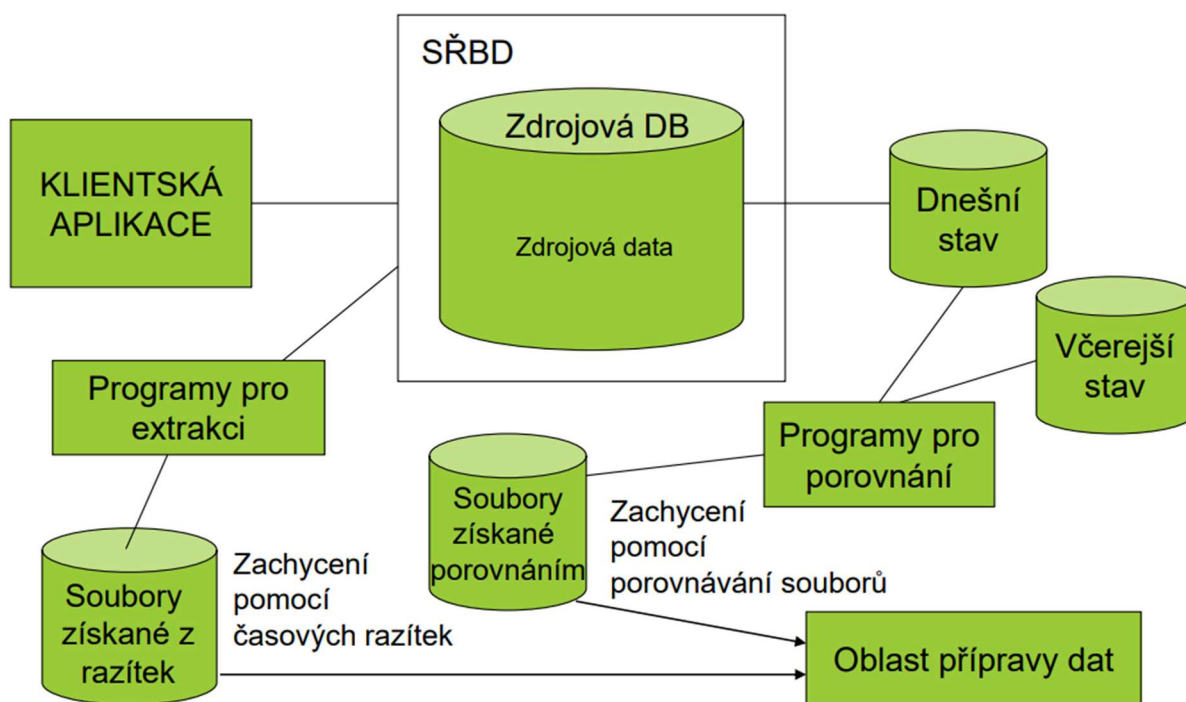


Obrázek č. 3 Proces přímé extrakce

(Zdroj:[4])

1.3.1.1.2. Metody odložené extrakce

Tyto metody nezachycují změny v databázi hned po změně, ale při nahrávání do databáze se porovnávají zdrojová a cílová databáze (Obrázek č. 4). Změny se zachycují pomocí časových razítek anebo pomocí porovnávání souborů. Razítky jsou označeny ty záznamy, které byly přidány anebo editovány. Razítko obsahuje například hodnotu od kdy do kdy ten záznam platí (VALID_FROM, VALID_TO). Porovnávání souborů je velmi neefektivní, vytvoří se soubor s kopií dat v aktuálním stavu a soubor s kopií s posledním aktuálním stavem, poté se soubory porovnají. [4]



Obrázek č. 4: Proces odložené extrakce

(Zdroj: [4])

1.3.2. Transformace

V transformační vrstvě musí dojít k očištění dat (odstranění duplicit, multiplicit) a k jejich vyčištění a dosažení potřebné kvality (vyloučení chyb, nepřesností atd.) [3, s. 445-448]. Tato část bývá finančně nejnáročnější a obvykle dosahuje až 60 % vynaložených nákladů [5]. Často je potřeba odstranit různé anomálie jako jsou chybějící znaky. Také je potřeba sjednotit a vyčistit různě rozdělené atributy (adresa jednou jako jeden řádek, podruhé jako tři řádky), různé datové typy pro stejné sloupce, různé formáty čísel a měn. [4]

1.3.3. Přenos dat (Loading)

Jedná se o přesun údajů a jejich uložení do tabulek datového skladu v určených časových periodách. Nahrávání dat by mělo probíhat automaticky. Většinou se jedná o časově náročnou operaci. Existují tři typy přenosů a několik módů. [3; 4]

Typy přenosů:

- Iniciální nahrávání – nahrání všech dat do prázdného datového skladu.
- Inkrementální nahrávání – promítnutí změn v databázi do datového skladu, probíhá periodicky.
- Přepis dat – kompletní smazání datového skladu a nahrání nových dat. [4]

Typy módů:

- Nahrání (Load) – pokud cílová tabulka obsahuje stejná data, pak jsou smazána a nahrazena aktuálními.
- Přidání (Append) – přidání nových dat k již zapsaným datům, při duplicitě uživatel vybírá další postup (zahodit nebo přepsat řádek).
- Destruktivní sloučení – stejné jako přidání s jediným rozdílem, při zjištění stejného řádku se přepíše hodnota řádku.
- Konstruktivní sloučení – při zjištění stejného řádku se přidá nový prvek, starý v datovém skladu zůstane. [4]

1.4. Datový sklad

V dnešním elektronickém světě je potřeba ukládat a shromažďovat velkou spoustu dat, proto se rozvinuly technologie datových skladů. Byl vytvořen pojem Data Warehousing, který definuje proces vytváření a užívání. Tato technologie je v dnešní době jeden z nejrozšířenějších a nejdůležitějších důvodů k rozvoji podniku a jeho informačního systému. [1, s. 32]

Definice pojmu podle jednoho ze zakladatelů Billa Inmona zní takto: „*Datový sklad (Data Warehouse) je integrovaný, subjektivě orientovaný, stálý a časově rozlišitelný souhrn dat, uspořádaný pro podporu potřeb managementu* [1].“

V této definici je definováno několik pojmů:

- Subjektově orientovaný – data jsou rozdělována podle jejich typu, ne podle aplikací, ve kterých vznikla. Data jsou uložena zaměstnancem pouze jednou, v produkčním systému jsou data rozptýlena podle toho, v které aplikaci jsou využívána.
- Integrovaný – data jsou ukládána na úrovni celého podniku, a ne podle oddělení.
- Stálý – datové sklady jsou typu „Read-Only“, ve kterých žádná data nevznikají a nelze je ani měnit. Data jsou načítána z externích zdrojů nebo databází.
- Časově rozlišený – v datovém skladu jsou uložena veškerá data, což znamená aktuální a historická, aby bylo možné provést analýzu za určité období, je k nim přiřazena časová dimenze. [1, s. 32]

1.5. Datové tržiště

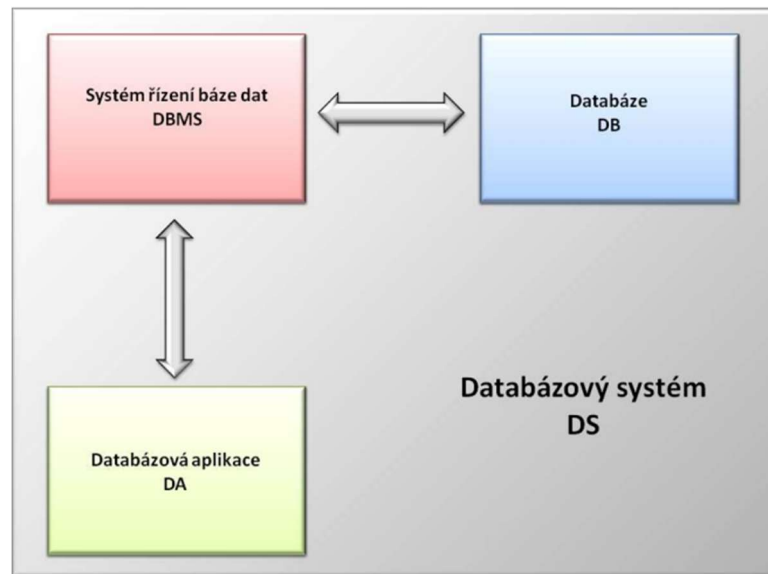
Datové tržiště neboli Data Marts, mají velmi podobný princip jako datové sklady. Největším rozdílem mezi datovým tržištěm a skladem je, že data z nich jsou určena jen pro část podniku a pro malý okruh uživatelů jako jsou jednotlivé divize, oddělení, pobočka nebo závod. Taková tržiště se posléze integrují do celopodnikového řešení = datového skladu. Výsledkem vytváření datových tržišť je zkrácení doby návratnosti investic, snížení nákladů a podstatné zmenšení rizika při integraci. [1]

1.6. Databáze

Pod pojmem databáze si můžeme představit soubor uspořádaných dat, který popisuje reálný svět jako například evidence studentů na vysoké škole anebo nákup zboží. V databázích jsou popsány entity a atributy. Entita je určitý prvek reálného světa jako například stroj nebo vyučovací předmět a atribut je vlastnost entity, respektive charakteristiky entity jako je například u stroje jeho název. Mezi entitami vytváříme v databázi vazby, které mezi sebou mají prvky v reálném světě. [6; 7]

Databázové systémy se skládají z několika částí (Obrázek č. 5). SŘBD (Systém řízení báze dat) je první částí databázového systému, která má na starosti organizování a udržování shromážděných informací z databázové aplikace. Databázová aplikace je další součástí databázového systému a je počítačový program, který pomocí jazyka SQL vyvolá příkaz pro SŘBD. Slouží k prohlížení, vybírání a aktualizování informací uložených pomocí SŘBD. Uložená data v databázi jsou poslední částí databázového

systemu. [6; 7]



Obrázek č. 5 Databázový systém

(Zdroj: [6])

1.6.1. Relační databáze

Relační databáze vychází z relačního modelu, který vznikl během 70. let minulého století. Relace je tabulka, která je tvořena ze sloupců a řádků. Sloupce se shodují s vlastnostmi entity a řádky zobrazují jednotlivé databázové záznamy neboli zobrazují aktuální stav světa. Jeden řádek v tabulce je nazván prvkem relace. Celá tabulka je nazývána relací a soubor těchto tabulek tvoří relační databázi. [7]

1.6.2. Jazyk SQL

SQL neboli Structured Query Language je databázový dotazovací jazyk, kterým definujeme data, dotazujeme se na ně a manipulujeme s nimi. Jazyk SQL vznikl v druhé polovině sedmdesátých let minulého století. V roce 1982 pracovní skupina v organizaci ANSI (American National Standards Institute) vytvořila první standardní verzi, tento standard byl několikrát přepracován a doplněn. V dnešní době je hojně využíván v podnicích pro snadné využití a velkou výkonnost. Je implementován v řadě databázových programů, ale v žádném není využívána plná standardní verze, nejvíce se standardu blíží jazyk SQL využívaný v MS SQL Serveru. [8, s. 103-105]

Příkazy jazyka SQL se dělí do následujících kategorií:

- DQL (Data Query Language) – nejpoužívanější a nejsložitější příkaz patřící do této kategorie je příkaz SELECT, který je určen pro výběr dat. Umožňuje vybírat sloupce z několika tabulek, řadit je a filtrovat zobrazené řádky pomocí podmínek např. WHERE, počítat souhrny (počet, průměr atd.).
- DML (Data Modification Language) – jsou příkazy pro práci s daty. Pomocí těchto příkazů je možné zapisovat a měnit uložené údaje v tabulkách, lze také přidat nebo odstranit záznamy. Příkazy v této skupině jsou INSERT (vkládání), DELETE (odstraňování), UPDATE (upravování).
- TCL (Transaction Control Language) – zvláštní skupina příkazů, které zajišťují bezpečnost uložených dat proti havárii například při výpadku proudu nebo celého systému. Při zabezpečené práci s tabulkou se provede buď celá skupina příkazů anebo v případě chyby se celá skupina neprovede.
- DDL (Data Definition Language) – jsou příkazy pro definici tabulek. Pomocí těchto příkazů je možné vytvářet tabulky (CREATE), upravovat stávající tabulky (ALTER), mazat tabulky (DROP) a pracovat s indexy.
- DCL (Data Control Language) – pomocí těchto příkazů lze vytvářet celé databáze, upravovat je. Nastavovat parametry databáze například použitou znakovou sadu, nebo spravovat přístup uživatelům. [8, s. 106-110]

1.7. OLAP

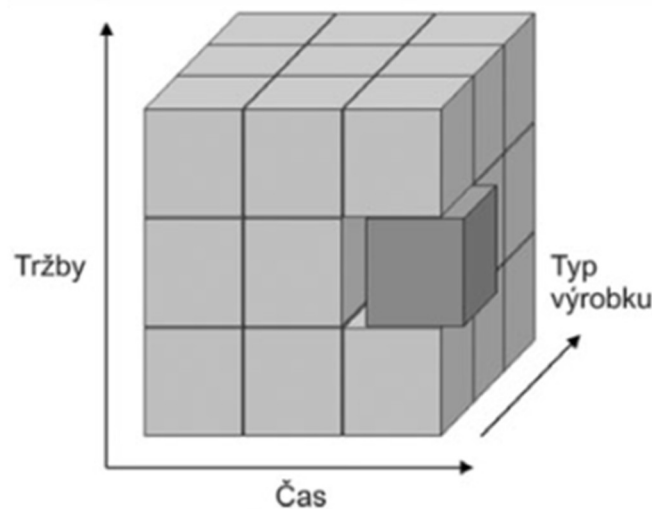
OLAP je zkratka anglického On-line Analytical Processing a označujeme jím nástroje pro rychlé zpracování analýz. OLAP je důležitou součástí nástrojů Business Intelligence. Pokud OLAP chápeme v širším kontextu, lze pracovat s pojmy „OLAP Server“ nebo „OLAP Databáze“, je předpokládáno, že datový sklad je součástí technologie OLAP. Jsou rozlišovány databáze OLTP (On-line Transaction Processing) a databáze OLAP. OLTP jsou v podstatě relační databáze, kde jsou data modifikována a uchovávána v podobě jednotlivých obchodních transakcí. Technologie OLAP může být součástí data-miningu nebo KDD (Knowledge Discover in Database, dobývání dat z databáze). OLAP na rozdíl od relačních databází pracuje s multidimenzionálními daty, které jsou uloženy ve formě vícerozměrných OLAP kostek. [9]

OLAP databáze je vytvořena z jedné nebo více OLAP kostek, které již obsahují

předzpracované agregace dat podle hierarchických struktur dimenzí a jejich kombinací.[5]

1.7.1. OLAP kostka

OLAP kostka obsahuje hodnoty ukazatelů z tabulky faktů agregované podle hierarchie definovaných dimenzí, kterou si můžeme představit jakožto datovou strukturu OLAP databáze. V OLAP databázi může existovat několik různých OLAP kostek s různými rozměry a úrovněmi agregace. Na obrázku je vidět příklad OLAP kostky (Obrázek č. 6). [1, s. 33]



Obrázek č. 6: OLAP kostka

(Zdroj: [3, s. 426-427])

1.7.2. Dimenze a fakta

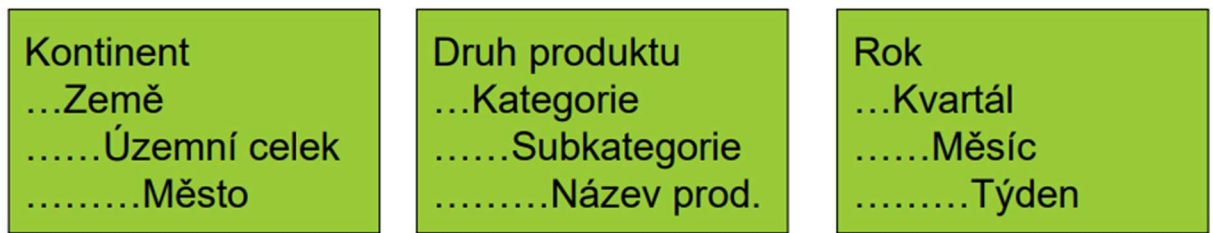
V každé datové OLAP kostce jsou obsaženy 2 typy údajů. [4]

1.7.2.1. Fakta

V databázi to bývá největší tabulka a obsahuje numerické měrné jednotky. Tato tabulka v kombinaci s dimenzemi vytváří určitá schémata. Tabulka faktů obsahuje cizí klíče do tabulky dimenzí, které rozšiřují význam dat v tabulce faktů. [4]

1.7.2.2. Dimenze

Logicky nebo hierarchicky uspořádané údaje, které jsou menší než tabulky faktů a nemění se tak často. Nejčastěji se jedná o časové, geografické či produktové dimenze. Tyto tabulky vedle zmíněných atributů také obsahují textové popisy obchodování. Na obrázku lze vidět několik příkladů dimenzí (Obrázek č. 7). [4]



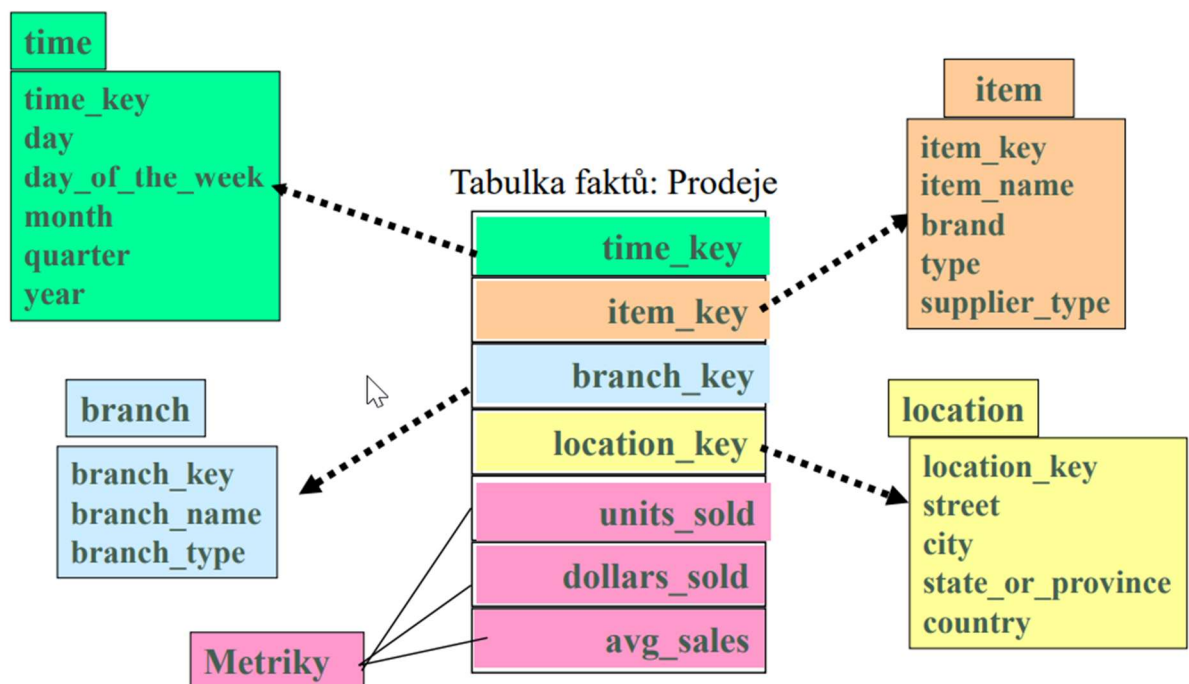
Obrázek č. 7: Příklad hierarchicky uspořádaných dimenzí
(Zdroj: [4])

1.8. Schémata

Vzájemným propojením tabulek faktů a dimenzí se vytváří různorodá schémata. V následující kapitole jich bude několik představeno.

1.8.1. Schéma hvězdy

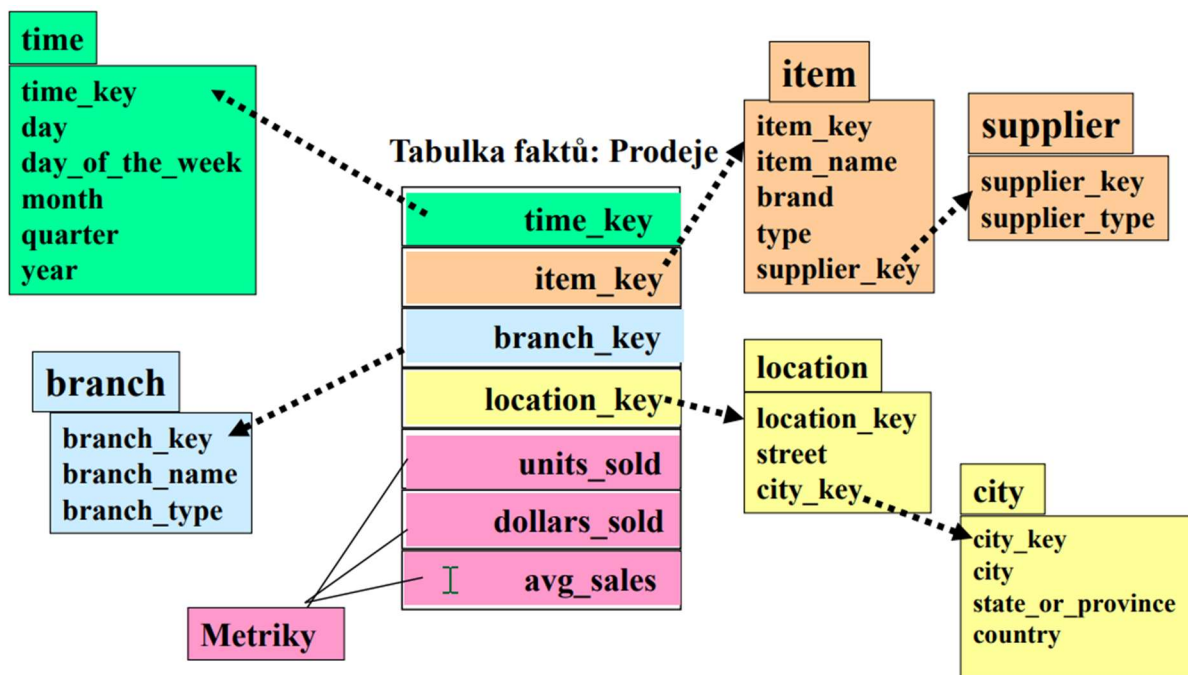
Tabulka faktů obsahuje cizí klíče, které jsou propojené do okolních dimenzí a poté se vztahují k jednotlivým primárním klíčům (Obrázek č. 8). Pro uživatele je toto schéma snadno čitelné. Při tomto schématu trvá delší dobu, než se nahrají data, ale jsou rychle čitelná a přístupná. [4]



Obrázek č. 8: Příklad schématu hvězdy
(Zdroj: [4])

1.8.2. Schéma sněhové vločky

Schéma sněhové vločky funguje na podobném principu jako schéma hvězdy s jediným rozdílem, že tabulky dimenzí se dále člení na další tabulky dimenzí a jsou složitě relačně svázány (Obrázek č. 9). Do databáze s tímto schématem se rychleji zapisují data, ale pomaleji se získávají. [10]



Obrázek č. 9: Příklad schématu sněhové vločky

(Zdroj: [10])

1.9. Microsoft Power BI

Jak již z názvu vyplývá tato aplikace byla vyvinuta společností Microsoft a patří do balíčku Microsoft 365. Vzhledem k velkému zastoupení Microsoft produktů ve firmě bude tato aplikace využívána k tvorbě reportů a vizualizací v této práci. Power BI je kolekce softwarových služeb, aplikací a konektorů, které spolupracují na přeměně nesourodých dat do přehledných vizualizací. [11]

1.9.1. Komponenty Power BI

Společnost Microsoft neustále investuje do rozvoje svých produktů, a proto v Power BI vznikají neustále nové komponenty pro lepší správu reportů s využitím umělé inteligence.

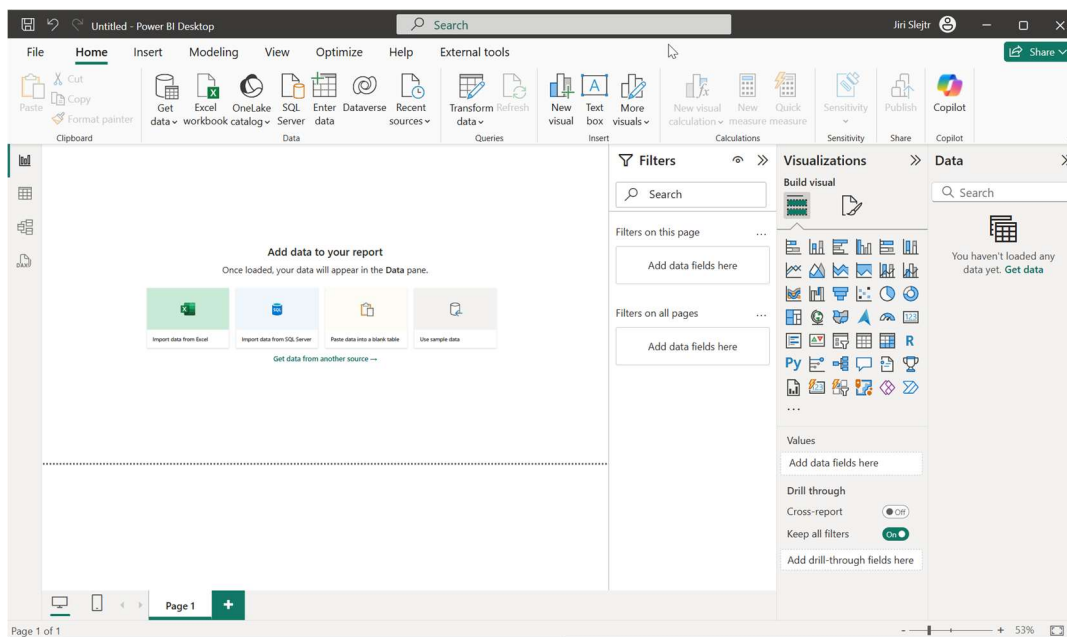
Mezi hlavní komponenty se řadí:

- Služba Power BI

Služba Power BI je software jako služba (SaaS) součástí Power BI, někdy označována také jako Power BI Online. Primární účel této služby je sloužit jako platforma, na které lze sdílet pracovní prostory s uživateli. V této službě lze také nasdílet vlastní připravené datové sady, aby je další uživatelé mohli využívat pro své vlastní vizualizace/sestavy. Pracovní prostory a datové sady lze seskupovat a publikovat ve formě aplikace, kam mají přístup jen příslušní lidé. [11]

- Power BI Desktop

Power BI Desktop je aplikace, která slouží k připojování k datům, transformaci dat a vizualizaci dat. Hlavními funkcemi této aplikace je proces ETL, vytváření vizualizací a sestav, spravování relací mezi tabulkami v datovém modelu. Proces ETL probíhá v Power Query, do kterého se dostaneme kliknutím na Transform Data (Obrázek č. 10). V aplikaci jsou k dispozici tři zobrazení – Sestava, Data a Model. V Sestavě se vytváří vizualizace, probíhá tam převážná část tvorby vizualizací. Na záložce Data se zobrazují tabulky, míry a ostatní data využitá v datovém modelu. Na záložce Model se spravují relace mezi tabulkami, popřípadě se vytváří nové. V aplikaci se dá též filtrovat vizuály, nastavovat vlastnosti vizuálů a přidávat do nich data. [11]



Obrázek č. 10: Prostředí aplikace Power BI Desktop

(Zdroj: Vlastní zpracování)

- Power BI Mobilní aplikace

Vzhledem k potřebě zobrazování vytvořených reportů z Desktop verze na mobilním zařízení byla k tomuto účelu vytvořena jednoduchá aplikace. Tato mobilní aplikace umožňuje také rychlé efektivní přidělování nebo odebrání přístupových práv k datovým sadám. [11]

Tyto tři hlavní komponenty jsou rozšiřovány podpůrnými komponenty a nástroji:

- Tvůrce sestav Power BI

Tvůrce sestav je nástroj Power BI, který umožňuje vytváření stránkových sestav (paginated reports). Paginated reports jsou výsledky sestav, které jsou ve formátu k jednoduchému tisku a sdílení, tzn. aby se všechny výsledky a vizualizace zobrazili jen na jedné stránce. Tento nástroj je primárně vhodný k vytváření faktur a štítků. [11]

- Power BI Embedded

Power BI Embedded je nástroj pro vývojáře aplikací, který umožňuje vkládat za pomoci programovacího jazyka JavaScriptu některé prvky Power BI do externě vyvíjených aplikací, což usnadňuje vývojářům práci. Tato služba je nedostupná v základní verzi Power BI, pro získání této služby je nutné mít předplatné Premium. [11]

1.9.2. Základní pojmy Power BI

- Kapacita

Základním prvkem konceptu Power BI jsou kapacity, což je sada prostředků, která se využívá k doručování obsahu Power BI. Konkrétně se jedná o procesor, paměť a úložiště. Rozlišujeme dva druhy kapacity, sdílená nebo rezervovaná. Sdílenou kapacitu si rozdělují mezi sebe uživatelé se standardní Power BI licenci, která je velmi omezena. Maximální velikost modelu je omezena na 1 GB a modely se mohou aktualizovat jen osmkrát denně. Naopak licence Power BI Premium používá rezervovaný druh kapacity, který umožňuje uživatelům až 400 GB paměti s využitím až 128 virtuálních jader. [11]

- Pracovní prostor

Pracovní prostor neboli Workspace, je místo kde se vytvářejí, publikují a spravují aplikace Power BI. Jde tedy o kontejner s veškerým obsahem, který je součástí aplikace Power BI, například datové sady nebo sestavy. [11]

- Aplikace

Aplikace je kolekcí řídicích panelů, sestav a sémantických modelů. Aplikace je výsledný produkt vytváření reportů publikovaný uživatelům. Aplikace lze sdílet i s uživateli bez licencí (Pro, Premium). Tyto aplikace jsou interaktivní, tzn. že se dají filtrovat, řadit atd., ale data nemůžou být upravovaná uživateli. Vlastník aplikace, může publikovat své sestavy pro určité firemní uživatele, kteří mají mít přístup do aplikace. [11]

- Toky dat

V případě většího projektu je občas zapotřebí propojit více zdrojů/souborů do jedné datové sady, k čemuž se využívají toky dat (Dataflow). Toky dat jsou ukládány jako entity v modelu CDM (Common Data Model). Velká výhoda je, že je lze importovat do jiných datových sad a externích nástrojů. [11]

- Datové sady

Datové sady (Dataset) jsou velké kolekce, ke kterým se lze připojit. Jedná se o základní prvek pro jednu nebo více sestav. Datové sady je možné sdílet ve více pracovních prostorech. Datové sady lze vytvořit z excelových souborů, které se nacházejí na OneDrive nebo SharePoint. [11]

- Sestavy

Sestavy Power BI obsahují jednu nebo více stránek, na kterých se nachází veškeré vizualizace, tabulky, grafy atd. Veškeré vizualizace v jedné sestavě vznikají z jedné datové sady. [11]

- Řídicí panely

Řídicí panel zobrazuje výčet pro firmu nejdůležitějších vizualizací z různých sestav. Největší výhodou řídicích panelů je přehledný monitoring nejdůležitějších metrik. [11]

1.9.3. Data Analysis Expressions (DAX)

DAX je jazyk vzorců používaný k vytváření vlastních výpočtů a měř v Microsoft Power BI a Excel Power Pivot. Jazyk DAX je založen na vzorcovém jazyku Excelu, je možné mezi těmito jazyky nalézt velké množství podobností. Rozdíl je v tom, že DAX nabízí mnohem větší flexibilitu při výpočtech u velkých datových sad. Vzhledem k jeho vlastnostem je důležitý nástroj pro datové analytiky, kterým umožňuje manipulovat s daty v tabulkách a analyzovat je efektivněji a s větší přesností. [11]

Jazyk DAX umožňuje vytvářet míry, což je jedna z nejdůležitějších funkcí jazyku DAX. Míry jsou výpočty, které lze využít k analýze dat různými způsoby. Pomocí míry lze například vypočítat celkovou průměrnou hodnotu efektivnosti zařízení. V aplikaci Power BI lze míry různě kombinovat mezi sebou a odkazovat se na ně. [11]

Další, co umožňuje jazyk DAX uživatelům je vytvářet kalkulované sloupce, které jsou velmi podobné mírám, ale přidávají do tabulky zcela nový sloupec. Kalkulovaný sloupec lze využít k vytvoření nových spojených sloupců anebo k vytvoření filtru. Tyto sloupce lze vytvářet pomocí stejných funkcionalit jako míry, což uživatelům umožňuje větší flexibilitu při analýze dat. [11]

1.10. Metody analýzy

V této kapitole se zaměřím na vybrané metody analýzy, které jsou často využívány při hodnocení a optimalizaci procesů, a které jsou využívány v této práci. Budu se věnovat SWOT analýze, Paretově analýze a PDCA cyklu.

1.10.1. SWOT analýza

SWOT analýza je komplexní analýzou vnějšího i vnitřního prostředí firmy. Slouží k vymezení silných stránek (Strengths), slabých stránek (Weaknesses), příležitostí (Opportunities) a hrozeb (Threats). Přínosy této analýzy jsou stručně a přehledně zobrazeny v tabulce nebo matici. Fakta pro analýzu SWOT lze získat pomocí různých technik, například využitím již vytvořených vlastních analýz anebo přímým porovnáváním s konkurentem (benchmarking). [12; 13]

Jednotlivé části SWOT analýzy se dají popsat takto:

Silné stránky – něco, v čem vynikáme a přináší nám to nějakou konkurenční výhodu oproti jiným firmám.

Slabé stránky – něco, co nás brzdí v rozvoji, věci, ve kterých firma ztrácí oproti konkurenci.

Příležitosti – příležitosti, které mohou zvednout poptávku po naší firmě a přinést firmě úspěch.

Hrozby – skutečnosti, které mohou snížit poptávku, negativní vliv na firmu. [13]

1.10.2. Paretova analýza

Paretova analýza je metoda, která stanovuje priority k odstranění hlavních problémů firmy. V roce 1897 ji definoval Vilfredo Pareto. Pareto ve svém pravidle vyvrátil, že vynaložené úsilí se rovná výsledkům. Analýza vychází z principu, že 20 % všech činností firmy přináší 80 % zisku firmy. Paretova analýza probíhá v 7 krocích:

1. Definování místa analýzy – výběr procesu nebo činností, kde firma chce zvýšit zisk nebo efektivitu, například snížit počet ztrát na výrobních strojích.
2. Sběr dat – získání relevantních dat, například data o ztrátách ze strojů.
3. Uspořádání dat – získaná data se seřadí podle určeného kritéria (počet ztracených kusů, počet výskytů atd.), řadí se vždy od největší po nejmenší.
4. Lorenzova kumulativní křivka – sečtou se všechny hodnoty u jednotlivých dat a zobrazí se v grafu.
5. Stanovení kritéria rozhodování – v tomto kroku se rozhoduje, jestli se bude striktně využívat Paretovo pravidlo anebo se sníží například, že chceme snížit jen 60 % ztrát.
6. Identifikování hlavních příčin – z levé strany grafu vzniklého v předchozím kroku, se vynese z hodnoty 80 % čára na Lorenzovu křivku, z které se poté udělá svislá čára, která oddělí ty příčiny ztrát (20 %), kterými se máme zabývat.
7. Stanovení nápravných opatření – vyhodnocení a stanovení příčin, které nám způsobují největší ztráty nebo vedou k navýšení zisku. [14]

1.10.3. PDCA cyklus

PDCA cyklus je iterativní metoda řízení, která je složena ze 4 kroků Plan-Do-Check-Act. Je využívána pro řízení podnikových nebo výrobních procesů. PDCA vychází z vědecké metody zkoumání problému (hypotéza, zkoumání, vyhodnocení). Slouží k systematické realizaci opatření, které se po zavedení analyzují a v případě úspěchu se standardizují. Jednoduchá struktura cyklu umožňuje opakovat sekvence rychle za sebou. Během opakování se stav mění s cílem dosáhnout požadovaného výsledku. Výhodou PDCA cyklu jsou malé a postupné změny, které šetří realizační náklady a snižují ztráty způsobené omylem. [15]

PDCA se jednoduše zavádí do praxe informováním zaměstnanců:

1. Plan (Plánuj) – definují se cíle, popíše se aktuální stav a vedoucí zformulují opatření, která vedou ke zlepšení.

2. Do (Delěj) – realizace nápravných opatření, sledování úspěšnosti zavedených opatření.
3. Check (Kontroluj) – zanalyzují se výsledky a srovnají se s počátečním cílem.
4. Act (Jednej) – pokud byly opatření úspěšná a cíl se podařilo splnit, cyklus končí a probíhá standardizace opatření. [15]

2. ANALÝZA SOUČASNÉHO STAVU

Tato část bakalářské práce je zaměřena na představení společnosti Bosch Powertrain s.r.o., ve které byl optimalizován a tvořen nový report v Power BI. Součástí této analýzy jsou základní informace o společnosti jako je historie, vlastnická struktura nebo popis společnosti, dále organizační struktura a aktuální stav využití Business Intelligence. Analýza současného stavu bude sloužit jako výchozí stav pro návrh řešení výstupu této bakalářské práce.



Obrázek č. 11: Logo koncernu Bosch

(Zdroj: Interní systém firmy)

2.1. Historie společnosti

Společnost Bosch má zastoupení po celém světě a působí v široké oblasti výrob. První zmínky o firmě pocházejí z konce 19. století. V roce 1886 založil Robert Bosch ve Stuttgartu dílnu pro jemnou mechaniku a elektrotechniku, ve které se zaměřoval na instalace telefonních systémů a elektrických zvonků. V roce 1897 začal do automobilů instalovat vlastní magnetoelektrické zapalování. V roce 1902 zapalování vylepšil hlavní inženýr firmy Bosch (Gottlob Honold) na vysokonapěťový systém magnetoelektrického zapalování se zapalovací svíčkou. Začátkem 20. století začala firma Bosch spolupracovat s Laurinem & Klementem a objevila se tak poprvé na českém území. [16]

V letech 1913-1914 po rozmachu automobilů firma uvedla na trh elektrický systém osvětlení, který obsahoval světlomety, generátor, regulátor a baterii. Kvůli první světové válce zmizely všechny zahraniční trhy. Výzkum a vývoj automobilových součástek byl

proto přerušen a Bosch přešel na výrobu válečných zbraní a výzbrojí, převážně se vyráběly detonátory ke granátům. Po skončení první světové války v roce 1917 čelila firma Bosch obrovským problémům, byly zrušeny veškeré patenty, které firma v té době měla. Poté se firma zaměřila na inovace a v roce 1926 přišla s osvětlením motocyklů a jízdních kol, houkačkou Bosch s pronikavým zvukem, stěrači a zapalování na bateriový pohon. V roce 1925 začala firma se zaváděním montážních linek, které dokázaly zkrátit dobu výroby magnetoelektrického systému zapalování z 50 na 5 dnů. V roce 1927 proběhla ve výrobě další inovace, která přinesla čerpadlo na vstřikování nafty, které v různých modifikacích přetrvalo až dodnes. První diesellové vstřikovací čerpadlo pro automobily bylo uvedeno na trh v roce 1936. V roce 1942 zakladatel společnosti Robert Bosch umírá a o tři roky později bylo zničeno více jak 50 % výrobních závodů společnosti. V polovině 50. let začala společnost Bosch vyvíjet elektronické komponenty, první z nich byla varioda, která se začala vyrábět v roce 1958. Následovaly tranzistory a integrované obvody. Během 50. let 20. století vstoupila společnost do oborů hydraulických systémů, lékařské elektroniky a radiotechniky. V roce 1963 koupila firma Bosch společnost specializovanou na balící stroje. Tím společnost rozšířila svou oblast působnosti. [17; 18; 19]

V roce 1995 byl uveden na trh elektronický stabilizační systém (ESP) a navigační systém TravelPilot s naváděním a hlasovým vstupem. V roce 1997 byl vynalezen systém common-rail – vysokotlaký injekční systém, který snížil spotřebu paliva u diesellových motorů, a v roce 2000 totéž následovalo, díky technologii DI Motronic (přímé vstřikování benzínu), u benzínových motorů. [20]

První oficiální pobočka na českém území byla založena roku 1920 v Praze na Letné. O pět let později byla firma přestěhována do vlastní budovy na třídu Maršála Focha (dnes Vinohradská třída) na pražských Vinohradech. Poté v roce 1945 byla firma okolnostmi donucena k přerušení výroby až do roku 1989, kdy začali s obnovou výrobních závodů v České republice. Od roku 1991, dceřiné společnosti firmy Robert Bosch GmbH, znovu vyrábějí na českém území. [21]

2.2. Vlastnická struktura

Vlastnická struktura společnosti Robert Bosch GmbH je poměrně neobvyklá, jak lze vidět na obrázku (Obrázek č. 12). Nezisková organizace Robert Bosch Stiftung GmbH

(Nadace Roberta Bosche) vlastní 94 % základního kapitálu společnosti Robert Bosch GmbH, ale nerozhoduje o žádných strategických nebo obchodních zaměřeních skupiny Bosch. Zbývající akcie vlastní společnost, která je ve vlastnictví rodiny Bosch (zhruba 5 %) a zbylé jedno procento vlastní Robert Bosch GmbH. V případě hlasovacích práv nejvíc drží společnost Robert Bosch Industrietreuhand KG přibližně 93 %. Zbývajících 7 % podílu drží rodina Bosch. [22]



Obrázek č. 12: Rozložení akcií a hlasovacích práv

(Zdroj:[22])

Nadace Roberta Bosche využívá prostředky z vlastnického podílu výhradně k dobročinným účelům. Nadace podporuje péči o zdraví, mezinárodní vztahy, sociální péči a vzdělávání.

2.3. Popis jihlavské společnosti

V roce 1993 byla v Jihlavě založena společnost s obchodním názvem BOSCH DIESEL s.r.o., která vystupuje od 1.1.2023 pod názvem Bosch Powertrain s.r.o. V prvopočátku měla společnost 160 zaměstnanců, postupně se stala celosvětově největším výrobním závodem pro moderní a inovativní dieselové vstříkovací systémy Common Rail v rámci Bosch Group. Od začátku konání společnosti Bosch Powertrain s.r.o. se zvedl stav zaměstnanců na 4 500 a společnost tak patří k největšímu zaměstnavateli i investorovi v Kraji Vysočina. Tato společnost má v Jihlavě tři závody, ve kterých vyrábí komponenty pro dieselové vstříkovací systémy Common Rail. Výrobky jako dieselová vysokotlaká vstříkovací čerpadla, vysokotlaké zásobníky (raily) a tlakové regulační ventily náleží k hlavním výrobkům společnosti. [23] „Od roku 2023 společnost rozšířila svůj výrobní program o oblast elektromobility a připravuje se na sériovou výrobu elektronáprav, elektrických motorů, statorů a rotorů pro elektrická

osobní vozidla [23].“

Celkově společnost Bosch Powertrain s.r.o. hraje důležitou roli v globálních aktivitách společnosti Robert Bosch GmbH. Poskytuje svým zákazníkům kvalitní technologie dieselových vstřikovacích systémů Common Rail.

2.4. Analýza vnitřního prostředí firmy

1. Strategie

Hlavní strategií firmy je dosáhnout maximálního zisku. Firma se toho snaží dosahovat inovacemi a konkurenční výhodou před ostatními. Vzhledem k tomu, že se jedná o velkou firmu jsou zavedeny kontrolní mechanismy výkonnosti.

2. Struktura

Vlastnická struktura je popsána v kapitole Vlastnická struktura (Kapitola 2.2.). Organizační struktura ve firmě je:

- obchodní a výrobní ředitel (jednatelé společnosti),
- vedoucí jednotlivých úseků (např. vedoucí personálního úseku, vedoucí jednotlivých výrob, vedoucí technických služeb atd.),
- vedoucí jednotlivých oddělení na úsecích,
- vedoucí pracovních skupin na odděleních,
- zaměstnanci a praktikanti.

3. Systémy

Pro své činnosti firma využívá více systémů. Každý systém obsahuje jiné typy informací a jednotlivé systémy spolu nejsou propojeny. Jednotliví zaměstnanci mají v systémech přehled a ví kde najít pro ně důležité informace, které potřebují k vykonávání své denní činnosti.

4. Styl vedení práce

Ve společnosti se využívá demokratický, horizontální styl řízení. Vedoucí zaměstnanci komunikují s podřízenými, ale rozhodující slovo má vedení. Při finálním rozhodnutí je brán zřetel na podřízené zaměstnance, kteří jsou odborníci v daných oblastech. Oficiální komunikace ve firmě probíhá pomocí MS Outlook. Pokud je potřeba něco doladit nebo upravit je možno využít méně formální komunikaci pomocí software MS Teams, ve kterém se plánují i týmové mítinky a schůzky.

5. Schopnosti

Zaměstnanci jsou odborníci na danou oblast, ve které působí. Zaměstnanci jsou pomocí školení a kurzů vedeni k neustálému rozvoji svých schopností a přizpůsobování novým technologiím a inovacím. Kvůli velké konkurenci firem v tomto odvětví je kladen na tento rozvoj velký důraz.

6. Spolupracovníci

Ve firmě pracují stovky stálých zaměstnanců a k tomu firma hojně využívá externích pracovníků. Většina těchto zaměstnanců se v práci nepotká, proto se zaměstnanci znají spíše v rámci jednotlivých oddělení a pracovišť. Mezi zaměstnanci panuje přátelská atmosféra. Kvalitní vztahy mezi zaměstnanci ve firmě Bosch zlepšují důvěru, otevřenou komunikaci a v neposlední řadě také výkonnost. Zaměstnanci v kancelářích mají možnost využívat home-office a někteří část týdne pracují z domu. Díky kvalitním vztahům mezi spolupracovníky probíhá komunikace a setkávání zaměstnanců i mimo pracovní prostředí.

7. Sdílené hodnoty

Firma klade vysoké nároky na pracovní nasazení, které se všichni zaměstnanci snaží dodržovat a vynakládají velké úsilí na úspěšný chod firmy. Za další sdílenou hodnotu by se dalo považovat chování k zákazníkům. Firma si zakládá na férovém a korektním jednání se zákazníky a vždy zaměstnanci dodržují pravidlo: „Zákazník na prvním místě.“ Tyto hodnoty a dobré dosavadní výsledky firmy zaměstnance motivují k naplňování těchto hodnot i do budoucna.

2.5. Popis současného stavu IS

Základními kameny informačního systému firmy jsou systémy od firmy SAP. Prvním z nich je ERP systém, ve kterém firma řídí veškeré důležité operace související s prodejem (vyřizování zákaznických objednávek, ukládání zákaznických dat, kontrola dodání ve stanovených lhůtách atd.). Vystavují se v něm také zakázky na náhradní díly, opravy, generální opravy, revize ke strojům. Dále v něm firma ukládá data o svých externích dodavatelích, vyřizuje v něm veškeré pohledávky od dodavatelů a ukládá archivní záznamy.

Druhým využívaným systémem je CRM systém pro řízení vztahů se zákazníky. V CRM

systemu se shromažďují veškerá data o zákaznících včetně kontaktních údajů. V systému je ukládána veškerá aktuální i minulá komunikace mezi firmou Bosch a zákazníkem.

V neposlední řadě jsou ve firmě využívány nástroje z balíčku MS Office. Komunikace mezi zaměstnanci, jak bylo zmíněno výše, probíhá převážně přes MS Outlook a MS Teams. Pro formálnější komunikaci mezi zaměstnanci a pro komunikaci se zákazníky je využíván MS Outlook, pro potřebu řešení okamžitých problémů si zaměstnanci volají přes MS Teams.

Ve firmě se dále využívá MES systém pro sledování výrobních procesů v reálném čase a jejich řízení. Z tohoto systému se dají zjistit aktuální informace o vyrobených kusech v daný den, prostojích, technických ztrátách a o celkové efektivitě zařízení (OEE).

Dále se ve firmě běžně využívá MS Excel, ve kterém jsou ukládány ručně zapisovaná data z výrobních strojů. Excel je využíván k vytváření jednoduchých databází. V neposlední řadě se ve firmě využívá Power BI pro vytváření reportů o výkonnosti podniku, technických ztrátách, stavu zakázek, stavu strojích atd.

Ve firmě je vytvořený Excel soubor s odkazy na 180 sledovaných výrob, které obsahují odkazy na umístění jednotlivých souborů obsahující výrobní ztráty a data. Excel se musí pravidelně udržovat, aby fungovaly všechny odkazy (Obrázek č. 13). Z důvodu občasného přesouvání souborů od jednotlivých výrob je udržování souboru časově náročné. Pro každou výrobu je vytvořený excel o 10 listech, který se jednou denně ručně aktualizuje. V Excelu je možné si zobrazit výrobní data z ručně zapisovaných dat od dělníků ve výrobě ve formě grafů, pareto analýzy, počtu vyrobených kusů atd. (Obrázek č. 14).

2.6. Analýza využití komponent BI

V současné době jsou ve firmě používány komponenty BI. S ohledem na velikost firmy je potřeba mít funkční proces tvorby pravidelných reportů výsledků.

Ve firmě jsou využívány Power BI reporty o stavu strojů, o spotřebě materiálů, o nákladech na údržbu a o výkonových ztrátách. V reportu o výkonových ztrátách je potřeba mít data v reálném čase, což doposud nebylo možné, protože jsou tvořeny excelovské exporty s daty na denní bázi. S přechodem na MES systém je možnost získávat data v reálném čase. Dále ve společnosti Bosch bude končit podpora Power BI Report Serveru, proto se stávající reporty přesouvají na Microsoft Power BI Cloud (Služba Power BI – Kapitola 1.9.1). Na kterém není možné pro pravidelnou aktualizaci mít připojený jakýkoliv Excel, který se využívá na zapisování výrobních ztrát a opatření k nim.

Tahle současná situace je výchozím bodem pro samotný praktický návrh řešení.

2.6.1. SWOT analýza reportovacích a analytických služeb

Poslední a závěrečnou částí analýzy bude SWOT analýza reportingu a analytických služeb firmy Bosch, které mají vliv na každodenní fungování firmy a jsou hlavním tématem práce. SWOT analýzu je možné využít k nejen obecnému fungování firmy, ale i na jeho dílčí segmenty.

Silné stránky

Silných stránek v této oblasti je k nalezení několik, firma využívá ke svému chodu informace z informačních systémů. Firma vytváří reporty v Power BI na denní bázi. Ve firmě je dobře strukturovaná datová základna, ke které se dá dobře přistupovat a získávat data. V této souvislosti se dá uvést, že vedení firmy vnímá tuto oblast jako velmi důležitou a vidí v ní přidanou hodnotu pro řízení společnosti. Firma má dostatek zaměstnanců, kteří se tomuto tématu mohou věnovat.

Slabé stránky

Mezi slabé stránky firmy se mohou zařadit závislost na každodenním ručním načítání dat z excelu, riziko špatného zápisu od zaměstnance ve výrobě a zapisování opatření v excelu. Z toho pramení další slabé stránky, v případě špatných dat bude ovlivněno

rozhodování vedoucích pracovníků firmy a hrozí finanční ztráty. Další slabá stránka věci je, že MES systém ještě není tolik prozkoumaný a přesně se neví, kde jsou uložena data. V neposlední řadě je slabinou využívání excelovských databází kvůli možnému neoprávněnému vniknutí a zcizení interních firemních dat.

Příležitosti

Příležitosti při zavedení analytických a reportovacích služeb ze systému MES může být v první řadě lepší podpoření rozhodovacích procesů než doteď. Dosud toto rozhodování probíhalo vždy po ukončení výrobního dne, u některých výrob po ukončení výrobního měsíce, s připojením na systém MES lze mít data v reálném čase. MES systém také může hodně zrychlit přístup k datům, jelikož je možné se připojit přímo k databázi a není potřeba připravovat data, například za pomoci Pythonu. V rámci MES systému je také možnost zjistit všechny údaje už na úrovni stanic, což je také velká pomoc pro vedení společnosti.

Dále je toto skvělá příležitost zbavit se neideálních excelů připojených k dosavadním reportům, protože se špatně omezuje, co lidé mohou zadat do Excel souboru a je velké riziko zcizení dat. Excel s opatřeními se nahradí tzv. ubladoo tabulkou naprogramovanou v PHP pro webové prostředí automaticky ukládající záznamy do tabulky v interní databázi.

Hrozby

Hrozby zavedení tohoto reportu na službu Power BI Cloud (Služba Power BI) mohou být spojené s obavou z něčeho nového. Jelikož zaměstnanci firmy nemají doposud s tímto moc zkušeností může nastat situace, kdy tuto změnu odmítnou, nebudou ji chtít využívat a nemusí v tom vidět urychlení a zlepšení své práce. Další hrozbou může být obava zaměstnanců z prodloužení prováděných úkolů.

Těmto hrozbám je třeba předcházet důkladnými pravidelnými školeními zaměstnanců, vysvětlováním nových funkcí. V opačném případě hrozí riziko zániku celého projektu.

V příložené tabulce (Tabulka č. 1) jsou shrnuty výstupy SWOT analýzy.

Tabulka č. 1: SWOT analýza

Silné stránky	Slabé stránky
<ul style="list-style-type: none"> • dobře strukturovaná datová základna • dostatek zaměstnanců, kteří se mohou věnovat vytváření reportů 	<ul style="list-style-type: none"> • každodenní ruční načítání dat do excelu • závislost na kvalitě a správnosti dat od zaměstnanců • nebezpečí excelovských databází
Příležitosti	Hrozby
<ul style="list-style-type: none"> • podpora manažerského rozhodování v reálném čase • zrychlení přístupu k datům • odpojení excelu od reportů • zlepšení řízení jednotlivých stanic 	<ul style="list-style-type: none"> • odpor zaměstnanců k zavádění nových věcí • obavy z časové náročnosti při vykonávání práce

(Zdroj: Vlastní zpracování)

3. VLASTNÍ NÁVRH ŘEŠENÍ

V této části bakalářské práce se zaměřím na postup, který byl využitý při vytváření reportu o výkonových ztrátách. Report by měl pokrýt výše zmíněné slabé stránky a příležitosti. Veškeré vizualizace a schémata jsou vytvořena v aplikaci Power BI, jak bylo zmíněno výše. Data byla získávána z Azure Databricks a z databází na SQL Serveru.

3.1. Požadavky

Prvním krokem pro zdárné splnění požadavků firmy byla schůzka s vedoucími pracovníky technického oddělení o potřebách využití Power BI reportu a poskytnutí datových zdrojů. Hlavní zobrazovaná data v Power BI reportu jsou zobrazena na 4 stranách. Na jedné straně jsou zobrazeny výkonové ztráty (organizační, technické, přeseřizení, kvalitativní a výkonové) na výrobních strojích ve formě Paretovy analýzy. Na dalších stranách jsou zobrazeny technické ztráty na jednotlivých výrobních strojích ve formě PDCA cyklu, celková efektivnost zařízení (OEE) a denní sledování podle linek.

Druhým krokem bylo důkladné prostudování poskytnutých dat a získání potřebných tabulek a sloupců, ze kterých byly vytvořeny dotazy. Bylo potřeba vytvořit tabulku v kódu PHP pro zapisování opatření k vysokým ztrátám, což není součástí této práce.

Třetím krokem byla transformace dat v Power Query, vypočítat hodnoty OEE a ostatních ztrát (organizační, kvalitativní, technické, přeseřizení a výkonové).

Posledním krokem bylo vytvoření požadovaných vizualizací a měř v jazyku DAX.

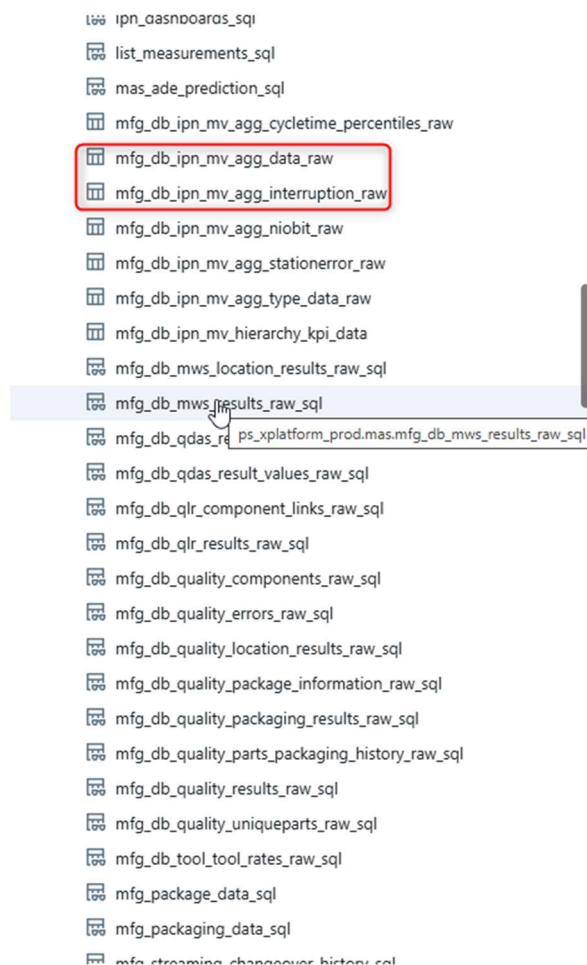
3.2. Datové zdroje

Před nahráním zdrojových dat do aplikace Power BI, ze kterých vycházely všechny vizualizace, bylo potřeba najít a zjistit, v jakých tabulkách se data nacházejí. Surová data, z datových skladů MES systému (Azure Databricks) a databází běžících na Microsoft SQL Serveru, bylo potřeba získat, očistit, upravit a připravit pro transformaci pomocí SQL dotazů.

Konkrétně byly využívány tyto tři zdroje dat:

- Azure Databricks

Pro správné fungování aktualizací reportu byly z tohoto datového skladu vytvořeny datové toky, protože v případě přímého napojení do Power BI nebyly data po publikaci reportu aktualizována automaticky. Pro účel vytvoření reportu byly využity dvě tabulky ze 48 možných (Obrázek č. 15). Databáze MES obsahuje data pro celý koncern Bosch tzn. ze všech výrobních závodů, proto je v každém dotaze vybírán závod Jihlava.



Obrázek č. 15: Zdrojové tabulky v Azure Databricks

(Zdroj: Interní systém společnosti)

Tabulka *ps_xplatform.mas.agg_data_raw*, která je na obrázku zobrazena jako *mfg_db_ipn_mv_agg_data_raw*, obsahuje již předpřipravená data o celkové výkonnosti strojů. V každém řádku jsou informace o OEE a jednotlivých ztrátách v sekundách po jednotlivých dnech a linkách, které bylo potřeba přepočítat na procentuální hodnoty. Na obrázku (Obrázek č. 16) je možné vidět část záznamů z výše zmíněné tabulky.

Tabulky obsahují velký počet sloupců, zobrazena je jen malá část.

IT	1.2 COUNT_PLAN	1.2 COUNT_SCRAP	1.2 COUNT_TARGET_NUREL_100	1.2 COUNT_TARGET_OEE	1.2 COUNT_TARGET_OEE_100	1.2 COUNT_TARGET_OEE_HIGH
1	0	0	0	0	0	0
2	0	0	0	5460	0	5460
3	0	5300	0	8545.454546	6836.363636	8545.454546
4	0	0	0	920.689656	764.172414	920.689656
5	0	800	0	411.428571	370.285714	411.428571
6	0	0	0	0	0	0
7	4276	3750	60	4645.16129	3414.193548	4645.16129
8	2345	3600	2	5166.666667	4236.666667	5166.666667
9	0	300	0	297.931034	297.931034	297.931034
10	0	800	0	0	0	0
11	1815	1613	14	3300	2508	3300
12	998	0	4	0	0	0
13	60	0	0	1004.651163	914.232559	1004.651163
14	0	0	0	920.689655	764.172414	920.689655
15	0	0	0	5094.339623	4584.90566	5094.339623
16	0	1700	0	2571.428571	2057.142857	2571.428571
17	3417	3500	0	4684.210528	3410.52632	4263.157896
18	0	0	0	2341.463415	0	0
19	259	0	0	289.932886	255.14094	289.932886
20	0	0	0	null	null	null
21	0	0	0	927.083334	769.479168	927.083334
22	0	4167	0	0	0	0
23	1007	0	0	1136.842105	1034.526316	1136.842105
24	254	0	3	275.159235	242.140127	275.159235

Obrázek č. 16: Příklad záznamů v tabulce `ps_xplatform.mas.agg_data_raw`

(Zdroj: Interní systém společnosti)

Tabulka `ps_xplatform.mas.agg_interruption_raw`, která je na obrázku zobrazena jako `mfg_db_ipn_mv_agg_interruption_raw`, obsahuje velký počet sloupců a každý řádek obsahuje záznam o jedné ztrátě, jak dlouho trvala, popis ztráty, na jaké lince a stanici se stala. Na obrázku (obrázek č. 17) je možné vidět část záznamů z výše zmíněné tabulky.

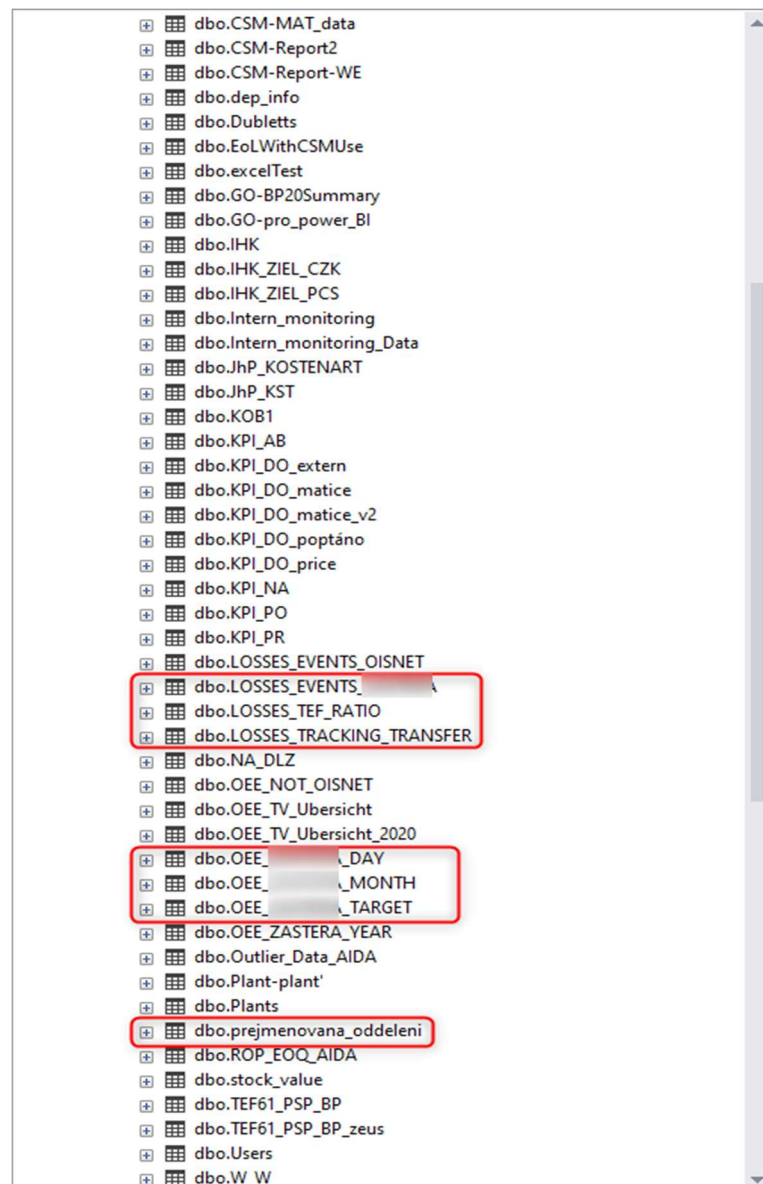
LOCATION_ID	I_CLASS	I_CLASS_DESCRIPTION	I_COUNT	I_FORMATTED_TYPE	I_GROUP	I_GROUP_DESCRIPTION	I_LOCATION_NAME
1	301	0 Planned	2	1000	1	Organizational	Line1001
2	302	0 Planned	2	1000	1	Organizational	Line1002
3	306	0 Planned	2	1000	1	Organizational	Line1006
4	01002000011	1 Unplanned	1	5	2	Technical	Line1.S120.S11.Fu1
5	013	0 Planned	1	1012	1	Organizational	Line2013
6	113	0 Planned	1	1008	1	Organizational	Line2113
7	213	0 Planned	1	1012	1	Organizational	Line2213
8	213	0 Planned	1	1008	1	Organizational	Line2213
9	312	0 Planned	1	1012	1	Organizational	Line2312
10	313	0 Planned	1	1012	1	Organizational	Line2313
11	313	0 Planned	1	1012	1	Organizational	Line2313
12	313	0 Planned	1	1008	1	Organizational	Line2313
13	003	0 Planned	1	1050	1	Organizational	Line2003
14	003	0 Planned	1	1001	1	Organizational	Line2003
15	101	1 Unplanned	1	2353	2	Technical	Line2101
16	103	1 Unplanned	1	1400	1	Organizational	Line2103
17	103	0 Planned	2	1000	1	Organizational	Line2103
18	103	1 Unplanned	1	2456	2	Technical	Line2103
19	103	1 Unplanned	1	1400	1	Organizational	Line2103
20	103	1 Unplanned	1	1064	1	Organizational	Line2103
21	103	1 Unplanned	2	2414	2	Technical	Line2103
22	103	0 Planned	1	1050	1	Organizational	Line2103
23	301	1 Unplanned	3	2763	2	Technical	Line2301
24	3030500	1 Unplanned	1	2862	2	Technical	Line2303.S6500

Obrázek č. 17: Příklad záznamů v tabulce `ps_xplatform.mas.agg_interruption_raw`

(Zdroj: Interní systém společnosti)

- SQL Server – Bluepuddle

Bluepuddle je interní server na kterém je několik databází. Pro účely mé bakalářské práce byly využívány 2 databáze: ForReporting a TefTools. ForReporting je databáze, do které se pomocí Python skriptu nahrávají zdroje z Excelu a rovnou se transformují do požadovaného tvaru. Z databáze ForReporting se využívá těchto 7 tabulek: *LOSSES_EVENTS_EXCEL*, *LOSSES_TRACKING_TRANSFER*, *LOSSES_TEF_RATIO*, *OEE_EXCEL_TARGET*, *OEE_EXCEL_DAY*, *OEE_EXCEL_MONTH* a *prejmenovana_oddeleni* (Obrázek č. 18).



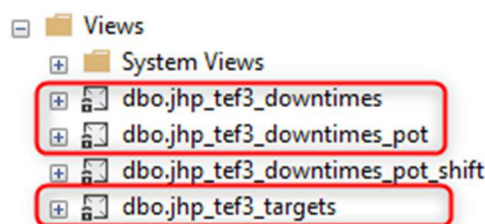
Obrázek č. 18: Databáze ForReporting v aplikaci Microsoft SQL Server

Z důvodu požadavku firmy o nezveřejňování přesných názvů tabulek, jsou názvy rozostřeny. (Zdroj: Interní systém společnosti)

Tabulka *LOSSES_EVENTS_EXCEL* obsahuje záznamy o ztrátách na strojích, které ještě nejsou převedené na MES nebo VIP systém a historická data před převedením. Tabulka *LOSSES_TRACKING_TRANSFER* obsahuje informace o jednotlivých linkách jako je datum, kdy byly převedené do MESu nebo VIPu. Tabulka *OEE_EXCEL_TARGET* obsahuje cíle, kterých má být dosaženo (kolik mají stroje za jednotlivý měsíc v roce vyrábět) a technické cíle (výše technických ztrát, které by stroj neměl přeskočit). Tabulky *OEE_EXCEL_DAY* a *OEE_EXCEL_MONTH* obsahují již spočítané procentuální hodnoty o výkonnosti strojů a ztrátách na denní, potažmo měsíční bázi. Tabulka *LOSSES_TEF_RATIO* obsahuje záznamy o technických ztrátách, kde bylo přítomno technické oddělení. Tabulka *prejmenovana_oddeleni* je překladová tabulka z důvodu změn názvů oddělení. Jako poslední se využívá tabulka *eKPI_opatreni* z databáze TefTools. Do této tabulky zapisují výrobní mistři opatření k technickým ztrátám na výrobních strojích.

- SQL Server – VIP databáze

VIP je externí databázi z firmy Robert Bosch s.r.o. z Českých Budějovic. V databázi máme přístup pouze k pohledům vytvořeným pro naše účely. Využívají se tři pohledy *jhp_tef3_downtimes*, *jhp_tef3_downtimes_pot* a *jhp_tef3_targets* (Obrázek č. 19). Pohled *jhp_tef3_downtimes* obsahuje záznamy o ztrátách strojů. *Jhp_tef3_downtimes_pot* obsahuje informace, jak dlouho linka mohla maximálně ten den vyrábět, z tohoto pohledu je poté možné vypočítat OEE v Power BI pro daný stroj nebo linku. Pohled *jhp_tef3_targets* obsahuje informace o cílech, kterých má být dosaženo.



Obrázek č. 19: Pohledy v databázi VIP v aplikaci Microsoft SQL Server

(Zdroj: Interní systém společnosti)

3.3. Načtení dat do Power BI

Dalším krokem v mé bakalářské práci bylo správné připojení datových zdrojů do Power BI. První byly připojeny datové toky z Azure Databricks. Tabulky (Kapitola 3.2) z tohoto datového zdroje obsahují velký počet sloupců, ze kterých bylo nutné vypočítat OEE

(z doby ztráty a celkového možného výrobního času za den). K získání potřebných dat byly vytvořeny 4 dotazy. Dva ze čtyř dotazů získávají a vypočítávají denní a měsíční OEE, jeden dotaz vypočítává cíle jednotlivých linek a poslední z těchto čtyř dotazů vybírá poruchy/ztráty na strojích. Denní OEE je počítáno pomocí vnořených dotazů a propojení obou používaných tabulek z MES databáze, jak je vidět na obrázku (Obrázek č. 20).

```

SELECT agg_data.WORKING_DAY, agg_data.IPN_MV_LOCATION_ID, agg_data.LOCATION_NAME, agg_data.pot, agg_data.oee, agg_data.ziel_oee,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else 100-Round((( NVL(agg_interruptions.ORGANISATIONAL,0) + NVL(agg_interruptions.TECHNICAL,0) + NVL(agg_interruptions.CHANGEOVER,0)
+ NVL(agg_data.time_nio,0) + NVL(agg_data.time_io,0))/NVL(agg_data.pot,0))*100,3) end AS oee_efficiency,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round((( NVL(agg_interruptions.ORGANISATIONAL,0) + NVL(agg_interruptions.TECHNICAL,0)
+ NVL(agg_interruptions.CHANGEOVER,0))/NVL(agg_data.pot,0))*100,3) end AS oee_availability,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round((NVL(agg_interruptions.ORGANISATIONAL,0)/NVL(agg_data.pot,0))*100,3) end AS oee_organizational,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round((NVL(agg_interruptions.TECHNICAL,0) / NVL(agg_data.pot,0))*100,3) end AS oee_technical,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round((NVL(agg_interruptions.CHANGEOVER,0) / NVL(agg_data.pot,0))*100,3) end AS oee_changeover,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round(((NVL(agg_data.time_nio,0)+NVL(agg_data.time_abort,0) )/NVL(agg_data.pot,0))*100,3) end AS oee_quality
FROM(
SELECT WORKING_DAY, IPN_MV_LOCATION_ID, LOCATION_NAME, Sum(TIME_TARGET_OEE_100) AS time_target_oee_100, Sum(TIME_TARGET_PROD_100) AS time_target_prod_100, Sum(TIME_TARGET_OEE_100) AS pot,
Sum(TIME_IO) AS time_io, Sum(TIME_ABORT) AS time_abort, Sum(TIME_TARGET_OEE) AS time_target_oee,Sum(TIME_NIO) AS time_nio,
CASE Sum(TIME_TARGET_OEE_100)
WHEN 0 THEN Round((Sum(TIME_IO)/Sum(TIME_TARGET_PROD_100))*100,3)
ELSE Round((Sum(TIME_IO)/Sum(TIME_TARGET_OEE_100))*100,3)
END AS oee,
CASE Sum(TIME_TARGET_OEE)
WHEN 0 THEN Round((Sum(TIME_TARGET_OEE)/Sum(TIME_TARGET_PROD_100))*100,3)
ELSE Round((Sum(TIME_TARGET_OEE) /Sum(TIME_TARGET_OEE_100))*100,3)
END AS ziel_oee
FROM ps_xplatform_prod.mas.mfg_db_ipn_mv_agg_data_raw
WHERE year >= YEAR(GETDATE())-2 AND IPN_MV_LOCATION_ID LIKE '5150%'
GROUP BY IPN_MV_LOCATION_ID, LOCATION_NAME, WORKING_DAY) agg_data

LEFT JOIN
( SELECT IPN_MV_LOCATION_ID, WORKING_DAY, COALESCE(ORGANISATIONAL,0) AS ORGANISATIONAL , COALESCE(TECHNICAL,0) AS TECHNICAL, COALESCE(CHANGEOVER,0) AS CHANGEOVER
FROM(
SELECT d.IPN_MV_LOCATION_ID, d.WORKING_DAY, sum(i.I_TIME_DURATION_MERGED) AS i_time_duration_merged,
CASE i_I_GROUP
WHEN '1' THEN 'organizational'
WHEN '2' THEN 'technical'
WHEN '3' THEN 'changeover'
ELSE 'na' END AS i_group
FROM
(
SELECT SHIFT_UUID, IPN_MV_LOCATION_ID, WORKING_DAY
FROM ps_xplatform_prod.mas.mfg_db_ipn_mv_agg_data_raw
WHERE year >= YEAR(GETDATE())-2 AND IPN_MV_LOCATION_ID LIKE '5150%' ) d
INNER JOIN ps_xplatform_prod.mas.mfg_db_ipn_mv_agg_interruption_raw i ON d.SHIFT_UUID = i.SHIFT_UUID
WHERE i.I_CLASS = '1'
GROUP BY d.IPN_MV_LOCATION_ID, d.WORKING_DAY, i_I_GROUP)
PRVOT( sum(i_time_duration_merged) FOR i_group IN ('organizational' AS ORGANISATIONAL, 'technical' AS TECHNICAL, 'changeover' AS CHANGEOVER))) agg_interruptions
ON agg_interruptions.IPN_MV_LOCATION_ID = agg_data.IPN_MV_LOCATION_ID AND agg_interruptions.working_day = agg_data.working_day
ORDER BY agg_data.working_day , agg_data.IPN_MV_LOCATION_ID

```

Obrázek č. 20: Dotaz na MES databázi k získání denní efektivity zařízení

(Zdroj: Vlastní zpracování)

Měsíční OEE je počítáno stejným způsobem jako dotaz k získání denního OEE zmíněného výše, jak je vidět na obrázku (Obrázek č. 21). Jediný rozdíl oproti dennímu OEE je, že měsíční OEE je seskupeno podle jednotlivých měsíců.

```

SELECT agg_data.WORKING_DAY_LOCAL_TS, agg_data.IPN_MV_LOCATION_ID, agg_data.LOCATION_NAME, agg_data.pot, agg_data.oee, agg_data.ziel_oee,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else 100*Round((( NVL(agg_interruptions.ORGANISATIONAL,0) + NVL(agg_interruptions.TECHNICAL,0) + NVL(agg_interruptions.CHANGEOVER,0)
+ NVL(agg_data.time_nio,0) + NVL(agg_data.time_io,0))/NVL(agg_data.pot,0))*100,3) end AS oee_efficiency,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round((NVL(agg_interruptions.ORGANISATIONAL,0) + NVL(agg_interruptions.TECHNICAL,0)
+ NVL(agg_interruptions.CHANGEOVER,0))/NVL(agg_data.pot,0))*100,3) end AS oee_availability,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round((NVL(agg_interruptions.ORGANISATIONAL,0)/NVL(agg_data.pot,0))*100,3) end AS oee_organisational,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round((NVL(agg_interruptions.TECHNICAL,0) / NVL(agg_data.pot,0))*100,3) end AS oee_technical,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round((NVL(agg_interruptions.CHANGEOVER,0) / NVL(agg_data.pot,0))*100,3) end AS oee_changeover,
CASE when (agg_data.pot = 0 or agg_data.pot IS NULL) then null else Round(((NVL(agg_data.time_nio,0)+NVL(agg_data.time_abort,0))/NVL(agg_data.pot,0))*100,3) end AS oee_quality
FROM (
SELECT TO_CHAR(WORKING_DAY_LOCAL_TS,'yyyymm') as WORKING_DAY_LOCAL_TS, IPN_MV_LOCATION_ID, LOCATION_NAME, Sum(TIME_TARGET_OEE_100) AS time_target_oee_100,Sum(TIME_TARGET_PROD_100) AS time_target_prod_100,
Sum(TIME_TARGET_OEE_100) AS pot, Sum(TIME_IO) AS time_io, Sum(TIME_ABORT) AS time_abort, Sum(TIME_TARGET_OEE) AS time_target_oee, Sum(TIME_NIO) AS time_nio,
CASE Sum(TIME_TARGET_OEE_100)
WHEN 0 THEN Round((Sum(TIME_IO)/Sum(TIME_TARGET_PROD_100))*100,3)
ELSE Round((Sum(TIME_TARGET_OEE)/Sum(TIME_TARGET_OEE_100))*100,3)
END AS oee,
CASE Sum(TIME_TARGET_OEE_100)
WHEN 0 THEN Round((Sum(TIME_TARGET_OEE)/Sum(TIME_TARGET_PROD_100))*100,3)
ELSE Round((Sum(TIME_TARGET_OEE) /Sum(TIME_TARGET_OEE_100))*100,3)
END AS ziel_oee
FROM ps_xplatform_prod.mas.mfg_db_ipn_mv_agg_data_raw
WHERE year >= YEAR(GETDATE())-2 AND IPN_MV_LOCATION_ID LIKE '5150%'
GROUP BY IPN_MV_LOCATION_ID, LOCATION_NAME, TO_CHAR(WORKING_DAY_LOCAL_TS,'yyyymm')) agg_data
LEFT JOIN
(SELECT IPN_MV_LOCATION_ID, WORKING_DAY_LOCAL_TS, COALESCE(ORGANISATIONAL,0) AS ORGANISATIONAL, COALESCE(TECHNICAL,0) AS TECHNICAL, COALESCE(CHANGEOVER,0) AS CHANGEOVER
FROM (
SELECT d.IPN_MV_LOCATION_ID, TO_CHAR(d.WORKING_DAY_LOCAL_TS,'yyyymm') as WORKING_DAY_LOCAL_TS, sum(i.I_TIME_DURATION_MERGED) AS i_time_duration_merged,
CASE i.I_GROUP
WHEN '1' THEN 'organisational'
WHEN '2' THEN 'technical'
WHEN '3' THEN 'changeover'
ELSE 'na'
END AS i_group
FROM (
SELECT SHIFT_UUID, IPN_MV_LOCATION_ID, WORKING_DAY_LOCAL_TS
FROM ps_xplatform_prod.mas.mfg_db_ipn_mv_agg_data_raw
WHERE year >= YEAR(GETDATE())-2 AND IPN_MV_LOCATION_ID LIKE '5150%') d
INNER JOIN ps_xplatform_prod.mas.mfg_db_ipn_mv_agg_interruption_raw i
ON d.SHIFT_UUID = i.SHIFT_UUID
WHERE i.I_CLASS = '1'
GROUP BY d.IPN_MV_LOCATION_ID, TO_CHAR(d.WORKING_DAY_LOCAL_TS,'yyyymm'), i.I_GROUP)
PIVOT( sum(i_time_duration_merged) FOR i_group IN ('organisational' AS ORGANISATIONAL, 'technical' AS TECHNICAL, 'changeover' AS CHANGEOVER))) agg_interruptions
ON agg_interruptions.IPN_MV_LOCATION_ID = agg_data.IPN_MV_LOCATION_ID AND agg_interruptions.WORKING_DAY_LOCAL_TS = agg_data.WORKING_DAY_LOCAL_TS
ORDER BY agg_data.WORKING_DAY_LOCAL_TS, agg_data.IPN_MV_LOCATION_ID

```

Obrázek č. 21: Dotaz na MES databázi k získání měsíční efektivity zařízení

(Zdroj: Vlastní zpracování)

Ztráty na strojích byly nahrány do datových toků pomocí níže zmíněného dotazu (Obrázek č. 22). Dotaz získávající ztráty z databáze MES je velmi jednoduchý oproti dotazům získávajícím OEE. V dotaze k získání ztrát jsou vybírány potřebné sloupce a jsou omezeny tak, aby ztráty nebyly starší než čtyři roky, nebyly plánovány a udály se v závodě Jihlava.

```

select INTERRUPTION_GROUP_DESCRIPTION as Category, LOCATION_LINE, DESCRIPTION as Komentar, TRIM(INTERRUPTION_CAUSE_DESCRIPTION) as Popis,
I_DURATION_SEC, I_START_LOCAL_TS as Zacatek, I_END_LOCAL_TS as Konec, LOCATION_STATION as Stanice, INTERRUPTION_TYPE_UUID, LOCATION_STATINDEX
from ps_xplatform_prod.mas.mfg_streaming_interruption_sql
WHERE year >= YEAR(GETDATE())-4 and I_CLASS_DESCRIPTION = 'Unplanned' and PLANT = 'JHP' and PERCENT = '100' order by I_START_LOCAL desc

```

Obrázek č. 22: Dotaz na MES databázi k získání výrobních ztrát

(Zdroj: Vlastní zpracování)

Poslední dotaz na MES databázi byl k získání cílů, jak dlouho by měl stroj denně vyrábět. V dotaze bylo potřeba cíl vypočítat pro jednotlivé linky ze dvou sloupců, vynásobit 100, protože celé OEE je zobrazováno v procentech a zaokrouhlit na tři desetinná místa, jak je vidět na obrázku (Obrázek č. 23).

```

SELECT WORKING_DAY_LOCAL_TS,
LOCATION_NAME,
CASE Sum(TIME_TARGET_OEE_100)
WHEN 0 THEN Round(((Sum(TIME_TARGET_OEE)/Sum(TIME_TARGET_PROD_100))*100,3)
ELSE Round((Sum(TIME_TARGET_OEE)/Sum(TIME_TARGET_OEE_100))*100,3)
END AS ziel_oee
FROM ps_xplatform_prod.mas.mfg_db_ipn_mv_agg_data_raw
WHERE PLANT_CODE = 5150
GROUP BY LOCATION_NAME,
WORKING_DAY_LOCAL_TS

```

Obrázek č. 23: Dotaz na MES databázi k získání výrobních cílů

(Zdroj: Vlastní zpracování)

Poté byly připojeny pohledy z databáze VIP, které byly popisovány v kapitole (Kapitola 3.2). VIP databáze byla připojena pomocí dvou dotazů. Jeden dotaz vybírá POT (maximální doba, kterou může stroj vyrábět za jeden den) ke každé lince pomocí kterého je počítáno OEE, jak je vidět na obrázku (Obrázek č. 24). Druhý dotaz vybírá ztráty na strojích. V dotaze jsou některé sloupce spojené do potřebného formátu a také jsou vypočítané počty kusů, které byly danou ztrátou ztraceny. Dotaz je omezený, aby vybíral data jen dva roky zpátky a nebyla linka, oddělení nebo produkt prázdné, tím jsou ošetřeny anomálie (Obrázek č. 25).

```
SELECT [department] as Department
      ,[NAZEV] as Line
      ,[product] as Product
      ,[day]
      ,[productionTime]
      ,[pot]
      ,[oee]
FROM [ENG_VIP_jhp].[dbo].[jhp_tef3_downtimes_pot] WHERE [product] IN ('CP3','Rotor') AND YEAR([day])>=YEAR(GETDATE())-2 AND [day] > {d '2020-08-01'}
```

Obrázek č. 24: Dotaz na VIP databázi k získání POT času

(Zdroj: Vlastní zpracování)

```
SELECT
[assemblyLineProduct] AS PRODUCT
,[assemblyLineDepartment] AS DEPARTMENT
,[assemblyLineName] AS LINE
,case when [workplaceName] IS NOT NULL THEN CONCAT([workplaceName], ' - ', ISNULL([workplaceDescription], '')) else null end as Station
,[timeFrom] AS 'Begin'
,[downtimeName]
,[codeCategoryLabel]
,case when [codeNumber] IS NOT NULL THEN CONCAT(ISNULL([codeCategoryPrefix], ''), [codeNumber], ' - ', ISNULL([codeLabel], '')) else [downtimeName] end as 'Description'
,[downtimeNote] as 'Comment'
,case when [downtimePieces] IS NULL AND [currentTtrCycleTime] >0 THEN ROUND([downtimeDuration]/[currentTtrCycleTime],1) ELSE [downtimePieces] end AS PCS
,[currentTtr]
,[downtimeDuration]
,[workday]
,cast(format([workday], 'yyyyMMdd') as int) AS [DATE]
,[downtimeTefWaitTime]
FROM [ENG_VIP_jhp].[dbo].[jhp_tef3_downtimes] WHERE [assemblyLineProduct] IN ('CP3','Rotor') AND YEAR([workday])>=YEAR(GETDATE())-2
AND [workday] > {d '2020-08-01'} AND ([assemblyLineDepartment] IS NOT NULL AND [assemblyLineName] IS NOT NULL AND [assemblyLineProduct] IS NOT NULL)
```

Obrázek č. 25: Dotaz na VIP databázi k získání výrobních ztrát

(Zdroj: Vlastní zpracování)

Nakonec byly připojeny databáze ForReporting a TefTools. Databáze ForReporting obsahuje data nahrané z excelu, které se zapisují ručně ve výrobě, a které ještě nepřešly na MES nebo VIP. Tabulky z ForReportingu byly připojeny pomocí následujících dotazů. Nejprve byla připojena tabulka *LOSSES_TRACKING_TRANSFER* pomocí jednoduchého výběrového dotazu, který vybere vše z tabulky. Tato tabulka obsahuje informace o linkách (produkt, oddělení, výroba, starý a nový název linky), které již přešly na systém MES nebo VIP a v následujících dotazech je tato tabulka využívána pro ošetření, aby se nevybírala data po datumu převedení. Dále byla připojena tabulka *prejmenovana_oddeleni*, ze které byly také vybrány všechny záznamy. Poté byla připojena tabulka *LOSSES_EVENTS_EXCEL*, v dotaze je spojené datum s časem začátku ztráty. Dotaz také obsahuje podmínku, aby se nevybírala data po převedení do nového systému a dále starší jak dva roky (Obrázek č. 26).

```

Select downtime.[PRODUCT]
,downtime.[DEPARTMENT]
,downtime.[PRODUCTION]
,[LINE]
,CONVERT(datetime,CONVERT(varchar(10),downtime.[DATE],120))+CAST([TIME] AS datetime) AS 'Begin'
,[TTNR]
,[STATION]
,[DESCRIPTION]
,[COMMENT]
,[PCS]
,[CODE]
,[CATEGORY]
,[CT]
,[DOWNTIME_DURATION]
,[TEF_PRESENT]
FROM [ForReporting].dbo.[LOSSES_EVENTS_
] as downtime
LEFT JOIN [ForReporting].[dbo].[LOSSES_
G_TRANSFER] as LineTransfer on downtime.PRODUCT = LineTransfer.PRODUCT and downtime.DEPARTMENT = LineTransfer.DEPARTMENT
and downtime.PRODUCTION = LineTransfer.PRODUCTION and downtime.LINE = LineTransfer.LINE_NAME
WHERE (LineTransfer.[DATE] is null or downtime.[DATE]<=LineTransfer.[DATE]) and (YEAR(CONVERT(datetime,CONVERT(varchar(10),downtime.[DATE],120))) = YEAR(GETDATE()))
or YEAR(CONVERT(datetime,CONVERT(varchar(10),downtime.[DATE],120))) = YEAR(GETDATE())-1 or YEAR(CONVERT(datetime,CONVERT(varchar(10),downtime.[DATE],120))) = YEAR(GETDATE())-2
or YEAR(CONVERT(datetime,CONVERT(varchar(10),downtime.[DATE],120))) = YEAR(GETDATE())-3)

```

Obrázek č. 26: Dotaz na tabulku LOSSES_EVENTS_EXCEL pro získání výrobních ztrát

(Zdroj: Vlastní zpracování)

Tabulka *OEE_EXCEL_DAY* byla připojena pomocí jednoduchého výběrového dotazu, který je omezen za pomoci tabulky *LOSSES_TRACKING_TRANSFER*, aby se vybíraly jen data do převodu, jak je vidět na obrázku (Obrázek č. 27).

```

Select oee.[PRODUCT]
,oe.[DEPARTMENT]
,oe.[PRODUCTION]
,oe.[LINE]
,oe.[OEE_TYPE]
,oe.[DATE] AS 'DATE'
,oe.[OEE]
FROM [ForReporting].dbo.[OEE_ZASTERA_DAY] as oee
LEFT JOIN [ForReporting].[dbo].[LOSSES_TRACKING_TRANSFER] as LineTransfer on oee.PRODUCT = LineTransfer.PRODUCT and oee.DEPARTMENT = LineTransfer.DEPARTMENT
and oee.PRODUCTION = LineTransfer.PRODUCTION and oee.LINE = LineTransfer.LINE_NAME
WHERE (LineTransfer.[DATE] is null or oee.[DATE]<=LineTransfer.[DATE])

```

Obrázek č. 27: Dotaz na tabulku OEE_EXCEL_DAY k získání denního OEE

(Zdroj: Vlastní zpracování)

Měsíční OEE je získáno z tabulky *OEE_EXCEL_MONTH* za pomoci jednoduchého dotazu s omezením stejným jako u denního OEE, jak je vidět na obrázku (Obrázek č. 28). V Power BI je do této tabulky napojena ještě druhá tabulka *LOSSES_TEF_RATIO*, která obsahuje poměr ke každé technické ztrátě, zda bylo nebo nebylo přítomno technické oddělení.

```

Select oee.[PRODUCT]
,oe.[DEPARTMENT]
,oe.[PRODUCTION]
,oe.[LINE]
,oe.[OEE_TYPE]
,oe.[DATE] AS 'DATE'
,oe.[OEE]
FROM [ForReporting].dbo.[OEE_
]MONTH] as oee
LEFT JOIN [ForReporting].[dbo].[LOSSES_TRACKING_TRANSFER] as LineTransfer on oee.PRODUCT = LineTransfer.PRODUCT and oee.DEPARTMENT = LineTransfer.DEPARTMENT
and oee.PRODUCTION = LineTransfer.PRODUCTION and oee.LINE = LineTransfer.LINE_NAME
WHERE (LineTransfer.[DATE] is null or oee.[DATE]<=LineTransfer.[DATE])

```

Obrázek č. 28: Dotaz na tabulku OEE_EXCEL_MONTH k získání měsíčního OEE

(Zdroj: Vlastní zpracování)

Po připojení tabulek s OEE bylo potřeba připojit tabulku *OEE_EXCEL_TARGET*, která obsahuje cíle, kterých má být dosaženo. Dotaz vybírá všechny sloupce z tabulky a poté ze sloupce TARGET spočítá průměr a vymění sloupce za řádky, aby cíle byly v řádcích

podle data a kategorie (Obrázek č. 29).

```
SELECT * FROM (
SELECT [PRODUCT]
      , [DEPARTMENT]
      , [LINE]
      , [CATEGORY]
      , [DATE]
      , [TARGET]
      , [PRODUCTION]
FROM [ForReporting].[dbo].[OEE_ZAS _ IGET])
as SrcTable PIVOT (AVG([TARGET]) for category in ("Technical","OEE")) as FinalTable
```

Obrázek č. 29: Dotaz na tabulku OEE_EXCEL_TARGET pro získání výrobních cílů

(Zdroj: Vlastní zpracování)

Z databáze TefTools se využívá jen tabulka eKPI_opatreni, ze které vybírám pomocí dotazu (Obrázek č. 30) všechny záznamy.

```
SELECT * FROM [TefTools].[dbo].[eKPI_opatreni]
```

Obrázek č. 30: Dotaz na tabulku eKPI_opatreni pro získání opatření k technickým ztrátám

(Zdroj: Vlastní zpracování)

3.4. Transformace dat pro vizualizaci v Power BI

Před vytvořením vizualizací v Power BI bylo potřeba data pocházející z více zdrojů (Bluepuddle, MES, VIP) upravit do jednotného formátu, v některých přidat chybějící sloupce, dopočítat OEE a poté jednotlivé datové zdroje spojit do sebe. Tabulky ze tří zdrojů (Bluepuddle, MES, VIP) obsahující výkonové ztráty, denní a měsíční OEE budou na závěr spojeny a využity k tvorbě reportů a vizualizací. Tabulky obsahující výkonové ztráty budou spojeny do jedné tabulky DOWNTIMES (dále již jen DT), tabulky obsahující denní OEE budou spojeny do OEE_DAILY_DATA (dále již jen ODD) a tabulky obsahující měsíční OEE budou spojeny do OEE_MONTHLY_DATA (dále již jen OMD), proto jsou upravovány samostatně. V reportu je ještě využívána kromě výše zmíněných statická tabulka CalendarYear, která slouží k zobrazení dat v grafech podle měsíců, historických hodnot a aktuálního průměru. Ke každému zdroji dat jsem připojil ještě cíle, kterých se má dosáhnout a tabulku *prejmenovana_oddeleni*, aby oddělení nesly správné názvy.

3.4.1. Tabulky s výkonovými ztrátami

- VIP Databáze

V prvním kroku jsem změnil datový typ datumu a počtu kusů. Potom jsem přidal vlastní sloupce, jestli byla ztráta plánovaná, rok, měsíc, den a dopočítal jsem konec ztráty ze začátku ztráty plus doby trvání ztráty. Přejmenoval jsem sloupce na požadované názvy, odstranil nepotřebné sloupce, přidal podmíněný sloupec pro přejmenování kategorie ztráty. Přidal sloupec s přejmenovanými výrobami a sloupec s rozlišením, jestli bylo přítomno technické oddělení. Poté jsem připojil tabulku *prejmenovana_oddeleni*, přejmenoval staré sloupce a sloučil nové názvy se starými, odebral nepotřebné sloupce, které byly využité na sloučení názvů oddělení a takto upravená data jsem připojil do spojené tabulky DT, která se využívá ve vizualizacích tohoto reportu.

- Bluepuddle

Nejprve jsem přidal tři vlastní sloupce, jeden pro stanici, aby před číslem stanice bylo vidět St., druhý pro přidání kategorie ztráty a třetí pro zjištění konce ztráty. Poté jsem původní sloupce, ze kterých jsem tvořil vlastní sloupce, odstranil. Následně jsem vyfiltroval řádky, které v popisu obsahovaly „Prázdné“. Poté jsem ještě přidal sloupce s rokem, měsícem a dnem. Takto upravená data byla připojena do datového zdroje DT pro vizualizace v reportu.

- MES

Prvním krokem bylo vytvoření sloupce ze slova Line a čísla linky, pomocí kterého jsem připojoval k názvu linky v MESu produkt, oddělení, výrobu a název linky, které se používají k běžnému označení z tabulky *LOSSESS_TRACKING_TRANSFER*. Poté jsem vyfiltroval záznamy, které nemají produkt (např. nesledované linky). Přidal jsem sloupce s rokem, měsícem a dnem. Ke sloupci stanice jsem přidal St. a připojil k ní další označení. V dalším kroku proběhlo odstranění nepotřebných sloupců. Z popisu ztráty bylo potřeba pro filtr vyříznout kódové označení ztráty. Takto připravená data jsem připojil k výše zmíněným do jedné zdrojové tabulky DT pro vizualizace v reportu.

3.4.2. Tabulky s denním OEE

- VIP Databáze

Z databáze VIP jsem musel denní OEE vypočítat za pomoci tabulky se ztrátami a sloupce POT (hodnota, jak dlouho stroj mohl za den vyrábět). Nejprve jsem přejmenoval sloupce, změnil datový typ u sloupce obsahující datum a čas jen na datum, seskupil řádky podle produktu, oddělení, výroby, linky, kategorie, datumu a spočítal jsem délku jednotlivých ztrát za jeden den. Poté jsem připojil výše zmíněný sloupec POT a spočítal hodnoty ztrátových veličin za pomoci vzorce:

$$OEE = \left(\frac{\text{Doba ztráty}}{POT} * 60 \right) * 100$$

Následně jsem OEE zaokrouhlil na tři desetinná místa. Poté jsem připojil tabulku s OEE, které bylo počítané v databázi. Odebral jsem již nepotřebný sloupec *doba ztráty* a přejmenoval sloupce na požadované názvy. Poté jsem připojil tabulku *prejmenovane_oddeleni* pro přejmenování oddělení na používané nové názvy. Poté jsem musel přidat dva vlastní sloupce na ruční přejmenování oddělení a výroby. Finálním krokem bylo připojení do společné tabulky ODD pro zobrazení ve vizualizacích v reportu.

- Bluepuddle

Prvním krokem bylo vyfiltrování anomálie v datech - 29.2.2023, která byla špatně zadána z výroby. Poté následovala změna datového typu datumu z textu na datum. Přejmenoval jsem sloupce, aby souhlasily názvy ve všech tabulkách a poté jsem přidat vlastní sloupec s kategorií a odstranil chyby ve sloupci *Product*. Finálním krokem bylo připojení do společné tabulky ODD pro zobrazení ve vizualizacích v reportu.

- MES

První krok v úpravě dat bylo vyfiltrovat testovací a nesledované linky, které pro tento report nebyly relevantní. Přejmenoval jsem vypočítané sloupce z dotazu na kategorie, připojil jsem produkt, oddělení, výrobu a název linky. Přidal jsem filtr k odstranění dat, které jsou před datumem převodu. Převodl jsem sloupce na řádky, přejmenoval jsem převedené sloupce. Seskupil řádky podle produktu, oddělení, výroby, linky, datumu a kategorie a spočítal součet OEE. Finálním krokem bylo opět připojení do společné tabulky ODD pro zobrazení ve vizualizacích v reportu.

3.4.3. Tabulky s měsíčním OEE

- VIP Databáze

Z databáze VIP jsem musel měsíční OEE vypočítat za pomoci tabulky se ztrátami a sloupce POT. Prvním krokem bylo vytvoření relativního datumu ve formě měsíc/Year-0. Poté jsem seskupil záznamy podle produktu, oddělení, výroby, linky, relativního datumu, kategorie a spočítal k nim celkový měsíční POT a sumu OEE. Následně jsem přidal tabulku se ztrátami a vyfiltroval plánované ztráty, aby v reportu zůstaly jen neplánované. Dále jsem přidal relativní datum, seskupil ztráty podle produktu, oddělení, výroby, linky, relativního datumu, kategorie a spočítal celkovou dobu v měsíci. Poté jsem připojil výše spočítaný měsíční POT a spočítal podle vzorce měsíční ztráty:

$$OEE = \left(\frac{\text{měsíční doba ztráty}}{\text{měsíční POT}} * 60 \right) * 100$$

Odebral jsem již nepotřebný sloupec obsahující celkovou dobu trvání, přejmenoval sloupec *PRODUKT* a *LINE_NAME* na *Product* a *Line*, aby ve všech datových zdrojích byly stejné názvy sloupců. Dalším krokem bylo spojení dvou připravených tabulek s výkonovými ztrátami a s OEE. Ve sloupci *Tef Present* jsem nahradil hodnoty *null* hodnotou „N/A“. Odebral jsem sloupec s měsíčním POT. Připojil jsem technické a OEE cíle z VIPu, přidal nový sloupec obsahující rok, pro který jsem použil výpočet (aktuální datum minus rok z relativního data). Připojil jsem tabulku *prejmenovana_oddeleni* pro správné názvy oddělení. Následně jsem pro větší přehlednost srovnal sloupce. Poté jsem změnil řádky sloupců kategorie a OEE na sloupce. Změnil jsem u sloupce *Technical* datový typ na *number*, poté jsem odstranil sloupec s výkonovými ztrátami (*Performance*) a dopočítal jsem je ve vlastním sloupci. Pokud hodnoty ve sloupci *OEE=null*, tak se sloupec *Performance=null*, pokud ne tak byl použit následující vzorec:

$$OEE = 100 - \sum OEE \text{ a ztrát}$$

Následujícím krokem bylo otočení sloupců na řádky, aby bylo OEE v jednom sloupci a ve druhém kategorie. Posledním krokem bylo připojení takto připravených dat do společné tabulky OMD zobrazované ve vizualizacích v reportu.

- Bluepuddle

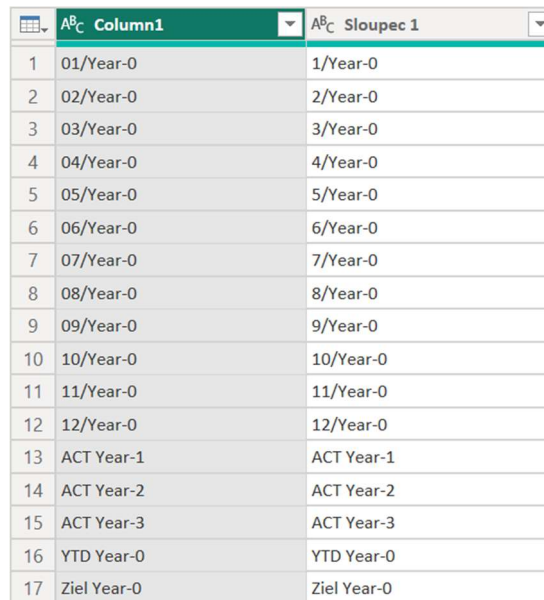
Prvním krokem bylo upravení dat z tabulky *LOSSES_TEF_RATIO*, kde jsem předělal datový typ datumu z textu na datum, poté jsem přidal sloupec s relativním datem. Dalším krokem bylo upravení dat z tabulky *OEE_EXCEL_MONTH*, ve které jsem prvně změnil datový typ datumu z textu na datum. Dále jsem přidal relativní datum, odebral sloupec s datumem, připojil jsem upravenou tabulku *LOSSES_TEF_RATIO*, přidal jsem vlastní sloupec *Category*, ve kterém jsem využíval data ze sloupce *OEE_TYPE* a poté jsem sloupec *OEE_TYPE* odebral. Následně jsem vytvořil nový sloupec *Tef Present* a přidal ke každé technické ztrátě dvě hodnoty „Yes“, „No“ a k ostatním řádkům hodnotu *null*. Rozbalil jsem hodnoty ve sloupci a vypočítal nové OEE následujícím způsobem: pokud sloupec *Tef Present* se rovná „Yes“, tak vynásob $OEE * \text{RATIO_YES}$, jinak použij $OEE * (1 - \text{RATIO_YES})$. Poté jsem odebral sloupec *OEE* a *RATIO_YES* a přejmenoval nově vypočítaný sloupec na *OEE*. V následujícím kroku jsem vytvořil sloupec *Year* (z aktuálního roku a čísla z relativního datumu). Poté jsem připojil technické a OEE cíle z Bluepuddle. Posledním krokem bylo připojení takto připravené tabulky do společné tabulky OMD využívané ve vizualizacích v reportu.

- MES

Nejprve jsem v úpravě dat vyfiltroval testovací linky, které pro tento report nejsou relevantní. Dalším krokem bylo přejmenování sloupců se ztrátami, získanými z dotazu, na správné názvy kategorií, tak jak jsou poté zobrazovány ve vizualizacích. Poté jsem převedl sloupce na řádky, přejmenoval převedené nové sloupce na kategorii a OEE. Připojil jsem správné názvy produktů, oddělení, výroby, linek a vyfiltroval jsem záznamy, které nemají přiřazený produkt. Vytvořil jsem datum přidáním 01 k měsíci a roku, který jsem měl k dispozici z databáze a toto jsem použil k vytvoření relativního datumu. Poté jsem přidal ke všem technickým ztrátám „Yes“, přítomnost technického oddělení. Dále jsem přidal sloupec, který obsahoval jen rok, pro připojení OEE cílů pro jednotlivé linky. Dále jsem připojil technické cíle z Bluepuddle, odebral jsem nepotřebné sloupce. Posledním krokem bylo připojení, takto připravené tabulky do společné tabulky OMD zobrazovaných ve vizualizacích v reportu.

3.4.4. Tabulka CalendarYear

Tabulka CalendarYear je ručně vytvořená tabulka, která na začátku obsahovala dva stejné sloupce každý pro jiný účel. Sloupec *Column1* je v reportu využíván pro zobrazení v grafech a sloupec *Sloupec1* je v reportu používán k propojení tabulek (Obrázek č. 31).



	Column1	Sloupec 1
1	01/Year-0	1/Year-0
2	02/Year-0	2/Year-0
3	03/Year-0	3/Year-0
4	04/Year-0	4/Year-0
5	05/Year-0	5/Year-0
6	06/Year-0	6/Year-0
7	07/Year-0	7/Year-0
8	08/Year-0	8/Year-0
9	09/Year-0	9/Year-0
10	10/Year-0	10/Year-0
11	11/Year-0	11/Year-0
12	12/Year-0	12/Year-0
13	ACT Year-1	ACT Year-1
14	ACT Year-2	ACT Year-2
15	ACT Year-3	ACT Year-3
16	YTD Year-0	YTD Year-0
17	Ziel Year-0	Ziel Year-0

Obrázek č. 31: Záznamy v tabulce CalendarYear

(Zdroj: Vlastní zpracování)

Přejmenoval jsem první sloupec na *RelativeDatePresent* a druhý sloupec na *RelDateDifferent*, pro pozdější využití pro výpočet měr v jazyku DAX. Poté jsem vytvořil sloupec *DatePresent* pro zobrazení hodnot v měsíčním grafu. Následně jsem vytvořil další sloupec *YearFilter*, který se využívá pro přepínání roků v zobrazení. Posledním krokem bylo připojení, za přiložený kalendář, kalendáře pro rok-1 a pro správné řazení sloupců v následných vizualizacích byl vytvořen sloupec *OrderColumn*, který přidává před hodnoty ACT 1, před měsíce 2 a před Ziel a YTD 3 (Obrázek č. 32).

	RelativeDatePresent	RelDateDifferent	DatePresent	YearFilter	orderByColumn
1	01/Year-0	1/Year-0	01/2025		2025 201/2025
2	02/Year-0	2/Year-0	02/2025		2025 202/2025
3	03/Year-0	3/Year-0	03/2025		2025 203/2025
4	04/Year-0	4/Year-0	04/2025		2025 204/2025
5	05/Year-0	5/Year-0	05/2025		2025 205/2025
6	06/Year-0	6/Year-0	06/2025		2025 206/2025
7	07/Year-0	7/Year-0	07/2025		2025 207/2025
8	08/Year-0	8/Year-0	08/2025		2025 208/2025
9	09/Year-0	9/Year-0	09/2025		2025 209/2025
10	10/Year-0	10/Year-0	10/2025		2025 210/2025
11	11/Year-0	11/Year-0	11/2025		2025 211/2025
12	12/Year-0	12/Year-0	12/2025		2025 212/2025
13	ACT Year-1	ACT Year-1	ACT 2024		2025 1ACT 2024
14	ACT Year-2	ACT Year-2	ACT 2023		2025 1ACT 2023
15	ACT Year-3	ACT Year-3	ACT 2022		2025 1ACT 2022
16	YTD Year-0	YTD Year-0	YTD 2025		2025 3YTD 2025
17	Ziel Year-0	Ziel Year-0	Ziel 2025		2025 3Ziel 2025
18	01/Year-1	1/Year-1	01/2024		2024 201/2024
19	02/Year-1	2/Year-1	02/2024		2024 202/2024
20	03/Year-1	3/Year-1	03/2024		2024 203/2024
21	04/Year-1	4/Year-1	04/2024		2024 204/2024
22	05/Year-1	5/Year-1	05/2024		2024 205/2024
23	06/Year-1	6/Year-1	06/2024		2024 206/2024
24	07/Year-1	7/Year-1	07/2024		2024 207/2024
25	08/Year-1	8/Year-1	08/2024		2024 208/2024
26	09/Year-1	9/Year-1	09/2024		2024 209/2024
27	10/Year-1	10/Year-1	10/2024		2024 210/2024
28	11/Year-1	11/Year-1	11/2024		2024 211/2024
29	12/Year-1	12/Year-1	12/2024		2024 212/2024
30	ACT Year-2	ACT Year-2	ACT 2023		2024 1ACT 2023
31	ACT Year-3	ACT Year-3	ACT 2022		2024 1ACT 2022
32	Ziel Year-1	Ziel Year-1	Ziel 2024		2024 3Ziel 2024

Obrázek č. 32: Výsledná podoba tabulky CalendarYear

(Zdroj: Vlastní zpracování)

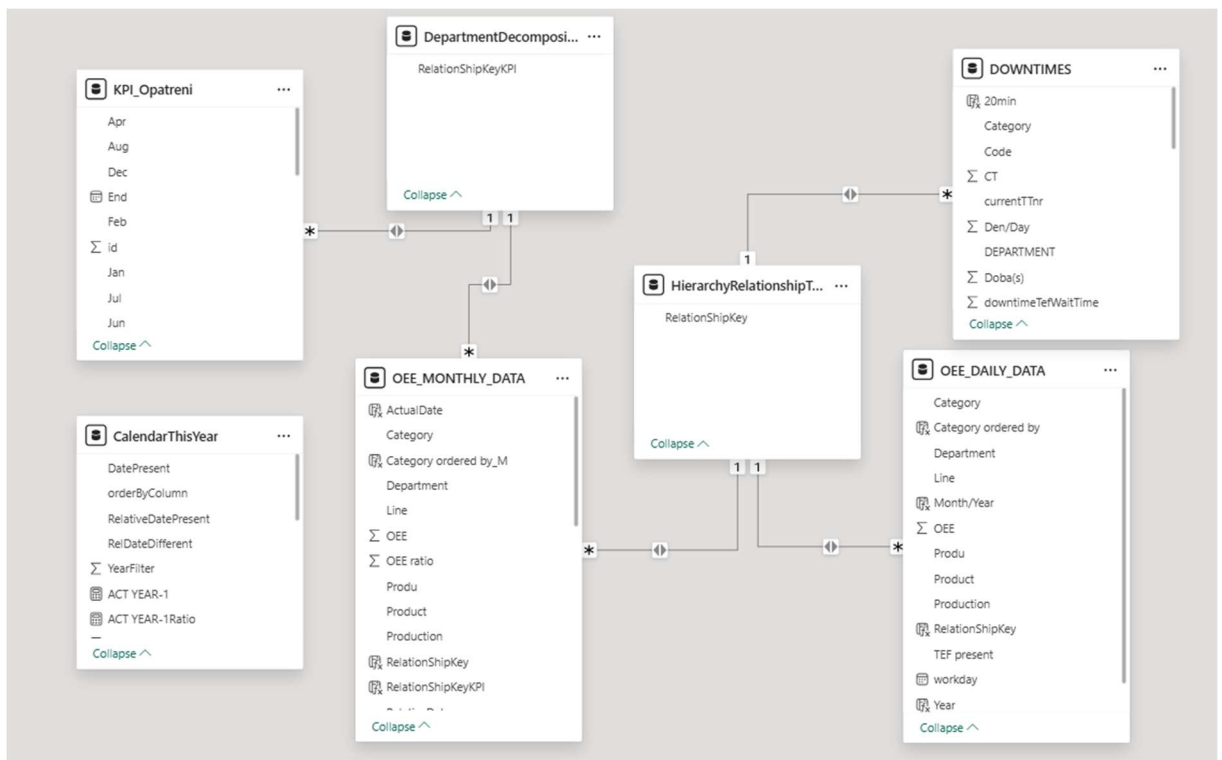
3.4.5. Tabulka KPI_Opatření

Prvním krokem bylo přejmenování všech sloupců, poté jsem vytvořil vlastní sloupec s rokem. Následně jsem vytvořil 12 vlastních sloupců, jeden pro každý měsíc, které slouží k zobrazení v tabulce podle datumu, jestli je plánovaný začátek/konec, jestli už byl začátek/konec a plánované nebo proběhlé milníky. Dále jsem vytvořil kopii sloupce Linka – stanice, který se do databáze zapisoval v jednom poli, abych mohl rozdělit linku a stanici od sebe pro pozdější využití při filtraci dat. Poté jsem nahradil hodnoty ve sloupci výroba. Takto upravená data byla využívána ve vizualizaci v reportu.

3.5. Vytvoření schéma datového modelu

Po připojení všech datových zdrojů (MES, VIP, Bluepuddle) bylo vytvořené následující schéma s relacemi mezi spojenými tabulkami (DT, ODD, OMD), jak je vidět na obrázku (Obrázek č. 33). Z toho se vychází v datovém modelu, kde jsou tři hlavní tabulky faktů

(DT, ODD, OMD) a jedna menší tabulka faktů KPI_Opatreni. Tabulky faktů jsou propojeny pomocí vytvořených tabulek (DepartmentDecomposition, HierarchyRelationshipTable) z důvodu vzniku relace M:N, které obsahují unikátní klíče vytvořené z produktu, oddělení, výroby a linky. Tabulka CalendarThisYear není propojena pomocí relace pro potřeby výpočtů, ale za pomoci výpočtu v mírách a funkce TREATAS.



Obrázek č. 33: Schéma datového modelu

(Zdroj: Vlastní zpracování)

3.6. Vytvořené míry a vizualizace

Z již vytvořené a připravené datové sady jsem podle požadavků firmy vytvořil míry v jazyce DAX. Celkově bylo vytvořeno v tomto reportu 29 měř. Výsledky těchto měř byly prezentovány ve formě specifických vizualizací jako jsou filtrované karty, tabulky, matice. V tomto reportu byly vytvořeny 4 stránky vizualizací. Stránka Top 10 Ztrát je výchozí stránka reportu a slouží k zobrazení všech výrobních ztrát. PDCA SMMC slouží ke sledování technických ztrát, na které se specializuje technické oddělení, pro které jsem report vytvářel. OEE Overview slouží k zobrazení efektivity výrobních strojů. Report Denní sledování je využíván k zobrazení výrobních ztrát podle linek a sečtení nejčastějších ztrát s možností filtrů po jednotlivých dnech nebo měsících. Příklad jedné

z vytvořených měř v tomto reportu (Obrázek č. 34) na podobném principu funguje většina měř v tomto reportu. Míry se odkazují na další výpočtové míry.

```

1 ValsForMatrix =
2 var firstChar = LEFT(SELECTEDVALUE(CalendarThisYear[RelDateDifferent]),1)
3 var res = IF(firstChar<>"A"&&firstChar<>"Y"&&firstChar<>"Z",if(CalendarThisYear
4 [CalculateOEENormalRatio]<>BLANK(),CalendarThisYear[CalculateOEENormalRatio],0),
5 IF(SELECTEDVALUE(CalendarThisYear[RelDateDifferent]) = "ACT Year-1",
6 [ACT YEAR-1Ratio],
7 IF(SELECTEDVALUE(CalendarThisYear[RelDateDifferent]) = "ACT Year-2",
8 [ACT YEAR-2Ratio],
9 IF(SELECTEDVALUE(CalendarThisYear[RelDateDifferent]) = "ACT Year-3",
10 [ACT YEAR-3Ratio],
11 IF(SELECTEDVALUE(CalendarThisYear[RelDateDifferent]) = "YTD Year-0",
12 [YTD YEAR-0Ratio],
13 IF(SELECTEDVALUE(CalendarThisYear[RelDateDifferent]) = "Ziel Year-0" ||
14 SELECTEDVALUE(CalendarThisYear[RelDateDifferent]) = "Ziel Year-1",
15 [Ziel Year-0-M],
16 "N/A"
17 )
18 )
19 )
20 )
21 RETURN res

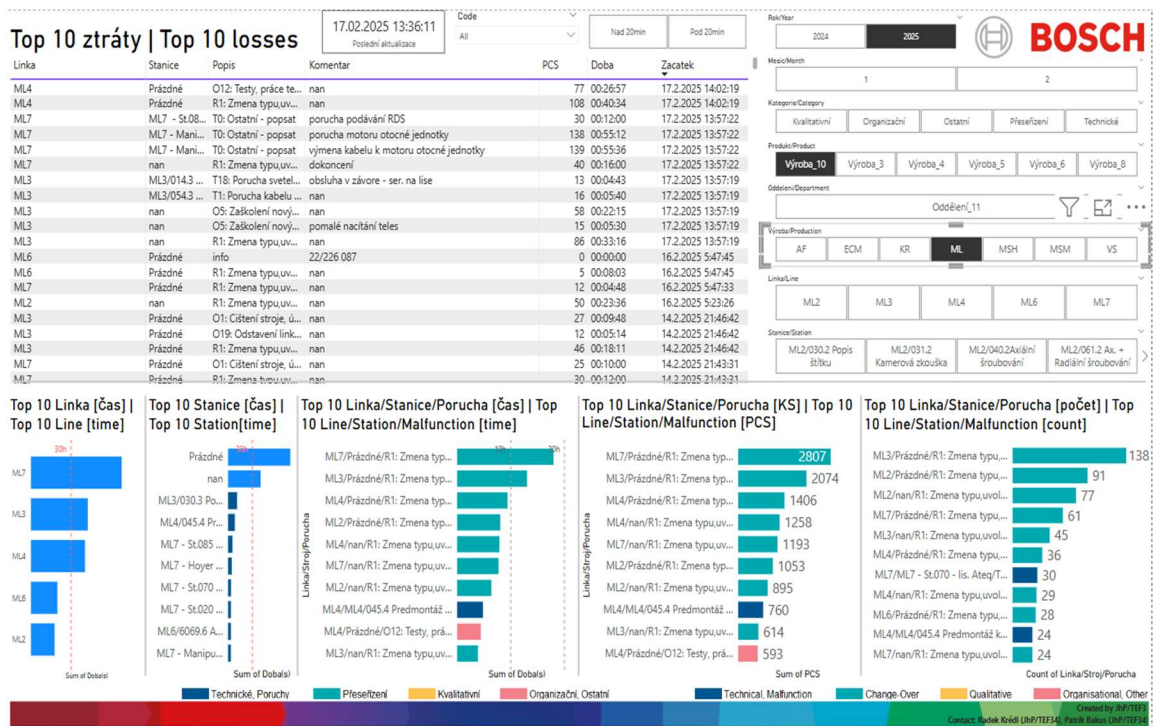
```

Obrázek č. 34: Syntaxe jazyka DAX u míry ValsForMatrix využívané v PDCA SMMC

(Zdroj: Vlastní zpracování)

3.6.1. Top 10 Ztrát report

V prvním reportu jsou využívány data ze spojené tabulky DT a zaměřuje se na zobrazení Paretovy analýzy, která obsahuje všechny ztráty podle data, jak následovaly za sebou, což lze vidět na obrázku (Obrázek č. 35). V levé horní části stránky je zobrazena tabulka s názvem linky, stanice, popis poruchy, komentář, ztracené kusy, dobu trvání a začátek poruchy. V pravé horní části stránky jsou zobrazeny průřezy (filtry) pro rok, měsíc, ztrátu, produkt, oddělení, výrobu, linku a stanici. Ve spodní části stránky je několik různých vizuálů. První zobrazuje stanice s nejčastějšími poruchami. Druhý zobrazuje stanice s nejdelšími poruchami. Třetí zobrazuje Top 10 nejčastěji se opakujících a nejdéle časově náročných poruch na stanicích. Čtvrtý graf zobrazuje Top 10 stanic a poruch, na kterých se ztratilo nejvíce kusů. Pátý graf ukazuje Top 10 stanic s nejčastěji opakujícími se poruchami.

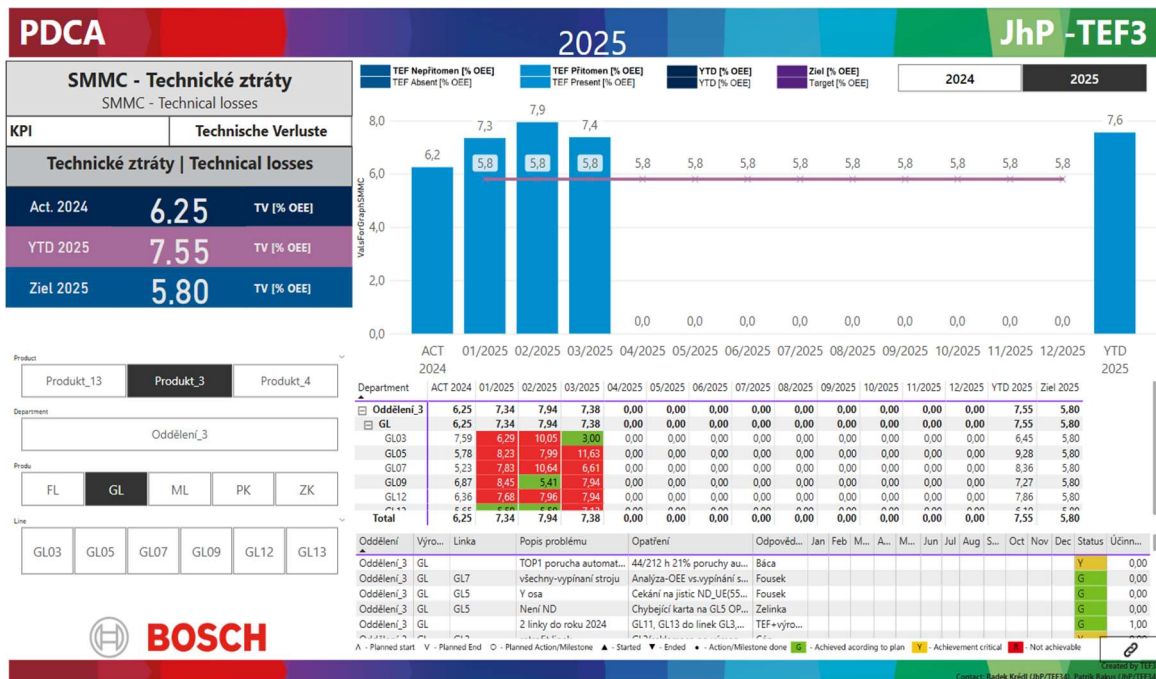


Obrázek č. 35: Report Top 10 Ztrát

(Zdroj: Vlastní zpracování)

3.6.2. Report PDCA SMMC

Druhá stránka vizualizací využívá data ze dvou tabulek OMD a KPIOPatření a zaměřuje se na systematiku PDCA. Na obrázku (Obrázek č. 36) v pravé horní části je zobrazen graf s technickými ztrátami podle měsíců, aktuální průměrná hodnota za celý rok a celková průměrná hodnota z loňského roku a cílová hodnota, které má být v tomto roce dosaženo. Tento report je stěžejní pro technické oddělení a pomáhá jim určovat opatření k vysokým technickým ztrátám. V grafu je vytvořena míra ValsForGraphSMMC, která vypočítává hodnoty do grafu za pomoci dalších měř, které vybírají hodnoty z minulého nebo aktuálního roku a průměrnou hodnotu letošního roku. Dále byla v této vizualizaci vytvořena matice za pomoci míry ValsForMatrix (Obrázek č. 34) s hodnotami podle oddělení, produkce a linky. Dole na stránce v této vizualizaci jsou zobrazená opatření, která mají zamezit vysokým technickým ztrátám a zapisují se do výše zmíněné tabulky. V levé spodní části stránky jsou přidány filtry na produkt, výrobu, produkci a linku. V pravém horním rohu je přepínač na výběr aktuálního nebo předcházejícího roku. V levé vrchní části stránky jsou vytvořené dlaždice, které obsahují aktuální hodnotu, cíle pro vybraný rok a poslední dlaždice zobrazuje hodnotu pro předcházející rok k vybranému roku.

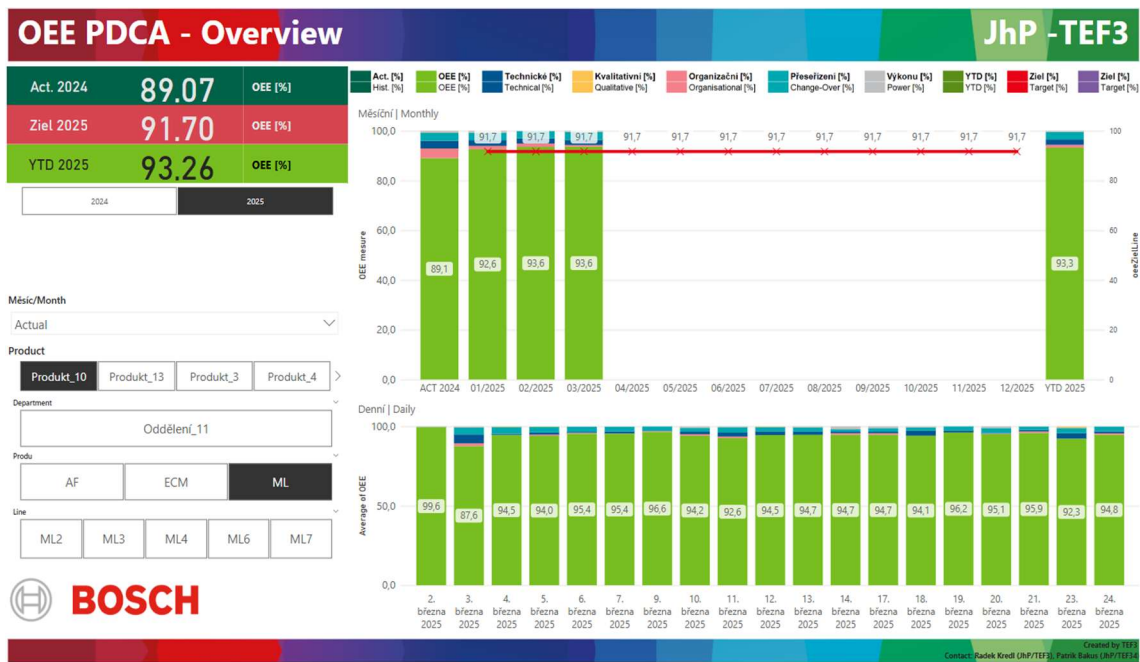


Obrázek č. 36: Report PDCA SMMC

(Zdroj: Vlastní zpracování)

3.6.3. Report OEE Overview

Třetí stránka vizualizací využívá data z tabulek ODD a OMD a slouží k zobrazení celkové efektivity strojů na denní a měsíční bázi, jak je vidět na obrázku (Obrázek č. 37). Tato strana vizualizací slouží hlavně pro hlavní a směnové mistry z výrob. V pravé horní části stránky je graf zobrazující hodnoty celkové výkonnosti strojů podle měsíců, je zobrazen taky loňský rok a průměrná hodnota aktuálního roku. Hodnoty v grafu jsou vypočítány pomocí míry *ValsForGraph* (míra funguje na stejném principu jako míra *ValsForMatrix* zobrazená na obrázku (Obrázek č. 34), která využívá další jednotlivé výpočty, jako jsou výpočty minulých let, aktuální hodnoty a průměrnou hodnotu OEE i všech ztrát za letošní rok. V dolní části je graf, který zobrazuje hodnoty dle vybraného měsíce. V levém horním rohu jsou zobrazeny dlaždice, které ukazují loňskou hodnotu OEE, aktuální hodnotu OEE a cíl.

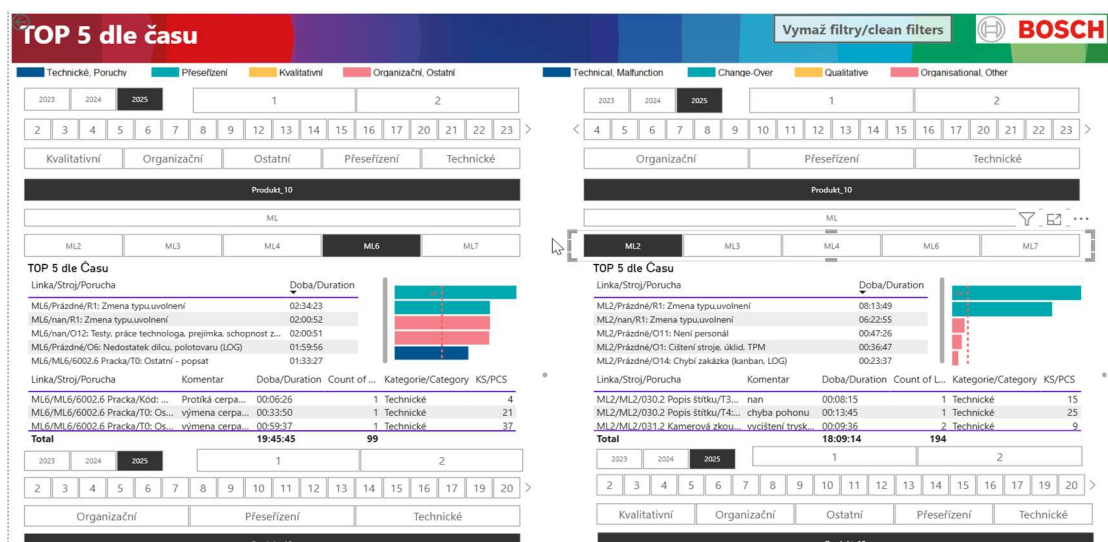


Obrázek č. 37: Report OEE PDCA Overview

(Zdroj: Vlastní zpracování)

3.6.4. Report denní sledování

Poslední vizualizace využívá tabulku DT a je vytvořena k porovnávání jednotlivých linek podle TOP 5 nejčastějších stanic a poruch, jak je vidět na obrázku (Obrázek č. 38). Tato strana obsahuje mnoho filtrů, jelikož byl požadavek mít možnost si vybrat až 8 linek a porovnat je proti sobě, popřípadě k nim přiřadit nějaké opatření. Stránka je posunovací, aby bylo možné zobrazit co nejvíc linek najednou.



Obrázek č. 38: Report Denní sledování

(Zdroj: Vlastní zpracování)

ZÁVĚR

Tato bakalářská práce se zabývá vytvořením reportu v aplikaci Power BI pro firmu Bosch Powertrain s.r.o. Cílem bylo vytvořit datovou základnu tak, aby se další příchozí výroby automaticky přidaly ze starého zdroje na zdroj MES nebo VIP, tento cíl se podařilo úspěšně splnit.

V první části byly vysvětleny teoretické pojmy potřebné k vypracování a pochopení návrhu řešení a také byly popsány některé nástroje Business Intelligence. Další část se věnovala aplikaci Power BI, ve které byl tvořen finální návrh řešení.

V druhé části byla představena společnost Bosch Powertrain s.r.o. Tato část obsahovala popis společnosti a její historii, poté byla provedena analýza vnitřního prostředí firmy, popis současného stavu informačního systému a využívání komponent Business Intelligence. Z těchto analýz a zjištění poté vycházel samotný výsledný návrh a report.

V poslední části této bakalářské práce byl popsán celý proces vytváření reportu v aplikaci Power BI. Nejprve byly zmíněny datové zdroje, ze kterých byly získány data obsahující informace o celkové výkonnosti strojů ve firmě. Data, ve formě jednotlivých tabulek faktů, byla následně transformována a upravena do potřebného tvaru podle následného využití například denní hodnoty OEE (Kapitola 3.4.2). Pro propojení a následné filtrování jednotlivých tabulek faktů bylo vytvořeno schéma datového modelu. Posledním krokem bylo vytvoření jednotlivých vizualizací, které poté byly publikovány do společného pracovního prostoru technického oddělení.

Výsledné řešení je v praxi využíváno pro informované rozhodování vedoucích pracovníků a snižování jednotlivých výkonových ztrát. Ve firmě tento report nahradil stávající excelové reporty, které se aktualizovaly jednou denně, u některých výrob jednou měsíčně. Aktualizace excelových reportů byla časově náročná, jelikož se musely ručně aktualizovat po jednotlivých linkách, popřípadě strojích. Report tvořený a popisovaný v této bakalářské práci je automaticky aktualizovaný a zobrazuje ztráty v reálném čase. Dále report šetří čas mistrů, kteří každé ráno museli aktualizovat výše zmíněné excely. Stal se standardním reportem, který využívá celá firma Bosch Powertrain s.r.o. v každodenní činnosti.

SEZNAM POUŽITÉ LITERATURY

- [1] NOVOTNÝ, Ota, Jan POUR a David SLÁNSKÝ. *Business intelligence: jak využít bohatství ve vašich datech*. Praha: Grada, 2005. ISBN 80-247-1094-3.
- [2] Hierarchie Data → Informace → Znalost. *Wikisofia* [online]. 2013 [cit. 2025-03-20].
Dostupné z: https://wikisofia.cz/wiki/Hierarchie_Data_%E2%86%92_Informace_%E2%86%92_Znalost#cite_note-petrikova-2
- [3] POUR, Jan, Miloš MARYŠKA, Iva STANOVSKÁ a Zuzana ŠEDIVÁ. *Self service business intelligence: jak si vytvořit vlastní analytické, plánovací a reportingové aplikace*. Grada Publishing, 2018. ISBN 978-80-271-0616-5.
- [4] BARTÍK, Vladimír. *Architektura a proces ETL - prezentace ke kurzu* [prezentace]. 2024 [cit. 2025-03-27].
- [5] POUR, Jan, Miloš MARYŠKA a Ota NOVOTNÝ. *Business intelligence v podnikové praxi*. Praha: Professional Publishing, 2012. ISBN 978-80-7431-065-2.
- [6] KŘÍŽ, Jiří. *Úvod do databázových systémů a metodologie jejich návrhu - prezentace ke kurzu* [prezentace]. 2023.
- [7] KŘÍŽ, Jiří a Petr DOSTÁL. *Databázové systémy*. Brno: Akademické nakladatelství CERM, 2005. ISBN 80-214-3064-8.
- [8] LAURENČÍK, Marek. *SQL: Podrobný průvodce uživatele*. Grada, 2018. ISBN 978-80-271-0774-2.
- [9] DANEL, Roman. *OLAP* [online]. 2025.
- [10] BARTÍK, Vladimír. *OLAP analýza - prezentace ke kurzu* [prezentace]. 2024.

- [11] *Dokumentace k Power BI* [online]. Copyright © Microsoft 2024 [cit. 2024-11-18]. Dostupné z: <https://learn.microsoft.com/cs-cz/power-bi/>
- [12] OTÁHAL, Martin. Co to je SWOT analýza? A k čemu slouží? *Co to je SWOT analýza? A k čemu slouží?* [online]. 2012 [cit. 2025-04-16]. Dostupné z: <https://rostecky.cz/co-to-je-swot-analyza-t2797>
- [13] PETR TYL, Jan. SWOT analýza. *SWOT analýza* [online]. 2017 [cit. 2025-04-16]. Dostupné z: <https://www.marketingmind.cz/swot-analyza/>
- [14] STŘELEČ, Jiří. Pareto analýza. *Pareto analýza* [online]. 2012 [cit. 2025-04-16]. Dostupné z: <https://vlastnicesta.cz/zkusenosti/vedeni-firmy/organizace-a-procesy/pareto-analyza/>
- [15] PDCA Cyklus. *PDCA Cyklus* [online]. 2025 [cit. 2025-04-16]. Dostupné z: <https://www.cems-cz.com/clanok/231-pdca-cyklus>
- [16] 1886-1905: Od první dílny k továrně. *Začátky v letech 1886–1905* [online]. 2024 [cit. 2024-12-08]. Dostupné z: <https://www.bosch.cz/novinky-a-pribehy/1886-1905-od-prvni-dilny-k-tovarne/>
- [17] 1906-1925: Globalizace a nový začátek. *Globální kroky 1906–1925* [online]. 2024 [cit. 2024-12-08]. Dostupné z: <https://www.bosch.cz/novinky-a-pribehy/1906-1925-globalizace-a-novy-zacatek/>
- [18] Auta a další vývoj v letech 1926–1945. *1926-1945: Od dodavatele pro automobilový průmysl ke specialistovi v oboru elektrotechniky* [online]. 2024 [cit. 2024-12-08]. Dostupné z: <https://www.bosch.cz/novinky-a-pribehy/1926-1945-od-dodavatele-pro-automobilovy-prumysl-ke-specialistovi-v-oboru-elektrotechniky/>
- [19] 1946-1959: Nový začátek a rekonstrukce. *Rekonstrukce v letech 1946–1959* [online]. 2024 [cit. 2024-12-08]. Dostupné z: <https://www.bosch.cz/novinky-a-pribehy/1946-1959-novy-zacatek-a-rekonstrukce/>
- [20] Počínaje rokem 1990: Odpovědi na výzvy globalizace. *Globalizace od roku 1990* [online]. 2024 [cit. 2024-12-08]. Dostupné z: <https://www.bosch.cz/novinky-a-pribehy/pocinaje-rokem-1990-odpovedi-na-vyzvy-globalizace/>

- [21] Bosch v České republice. *Bosch v České republice* [online]. 2024 [cit. 2024-12-08].
Dostupné z: <https://www.bosch.cz/nase-spolecnost/bosch-v-ceske-republice/>
- [22] Bosch Group ve světě. *Skupina Bosch ve zkratce* [online]. 2024 [cit. 2024-12-08].
Dostupné z: <https://www.bosch.cz/nase-spolecnost/bosch-group-ve-svete/>
- [23] Jihlava. *Bosch* [online]. 2024 [cit. 2024-12-08]. Dostupné z:
<https://www.bosch.cz/nase-spolecnost/bosch-v-ceske-republice/jihlava/>
- [24] LACKO, Luboslav. *Databáze: datové sklady, OLAP a dolování dat s příklady v Microsoft SQL Serveru a Oracle*. Brno: Computer Press, 2003. ISBN 8072269690.

SEZNAM OBRÁZKŮ

Obrázek č. 1: Obecná koncepce architektury BI	14
Obrázek č. 2: Vztahy mezi daty, informacemi a znalostmi	16
Obrázek č. 3 Proces přímé extrakce.....	17
Obrázek č. 4: Proces odložené extrakce	18
Obrázek č. 5 Databázový systém.....	21
Obrázek č. 6: OLAP kostka	23
Obrázek č. 7: Příklad hierarchicky uspořádaných dimenzí	24
Obrázek č. 8: Příklad schématu hvězdy.....	24
Obrázek č. 9: Příklad schématu sněhové vločky	25
Obrázek č. 10: Prostředí aplikace Power BI Desktop.....	26
Obrázek č. 11: Logo koncernu Bosch.....	32
Obrázek č. 12: Rozložení akcí a hlasovacích práv	34
Obrázek č. 13: Zdrojový excel s odkazy na výrobní data.....	38
Obrázek č. 14: Soubor s OEE a ztrátovými veličinami	38
Obrázek č. 15: Zdrojové tabulky v Azure Databricks	43
Obrázek č. 16: Příklad záznamů v tabulce ps_xplatform.mas.agg_data_raw	44
Obrázek č. 17: Příklad záznamů v tabulce ps_xplatform.mas.agg_interruption_raw	44
Obrázek č. 18: Databáze ForReporting v aplikaci Microsoft SQL Server	45
Obrázek č. 19: Pohledy v databázi VIP v aplikaci Microsoft SQL Server.....	46
Obrázek č. 20: Dotaz na MES databázi k získání denní efektivnosti zařízení	47
Obrázek č. 21: Dotaz na MES databázi k získání měsíční efektivnosti zařízení.....	48
Obrázek č. 22: Dotaz na MES databázi k získání výrobních ztrát	48
Obrázek č. 23: Dotaz na MES databázi k získání výrobních cílů.....	48
Obrázek č. 24: Dotaz na VIP databázi k získání POT času	49

Obrázek č. 25: Dotaz na VIP databázi k získání výrobních ztrát	49
Obrázek č. 26: Dotaz na tabulku LOSSESS_EVENTS_EXCEL pro získání výrobních ztrát	50
Obrázek č. 27: Dotaz na tabulku OEE_EXCEL_DAY k získání denního OEE	50
Obrázek č. 28: Dotaz na tabulku OEE_EXCEL_MONTH k získání měsíčního OEE... ..	50
Obrázek č. 29: Dotaz na tabulku OEE_EXCEL_TARGET pro získání výrobních cílů ..	51
Obrázek č. 30: Dotaz na tabulku eKPI_opatreni pro získání opatření k technickým ztrátám	51
Obrázek č. 31: Záznamy v tabulce CalendarYear	56
Obrázek č. 32: Výsledná podoba tabulky CalendarYear	57
Obrázek č. 33: Schéma datového modelu.....	58
Obrázek č. 34: Syntaxe jazyka DAX u míry ValsForMatrix využívané v PDCA SMMC	59
Obrázek č. 35: Report Top 10 Ztrát.....	60
Obrázek č. 36: Report PDCA SMMC	61
Obrázek č. 37: Report OEE PDCA Overview	62
Obrázek č. 38: Report Denní sledování	62

SEZNAM TABULEK

Tabulka č. 1: SWOT analýza.....	41
---------------------------------	----