

ZÁPADOČESKÁ UNIVERZITA V PLZNI

FAKULTA STROJNÍ

Studijní program: B 2301 Strojní inženýrství
Studijní zaměření: Průmyslové inženýrství a management

BAKALÁŘSKÁ PRÁCE

Plánování výroby pomocí Advanced Planning Scheduling

Autor: **Jan Průcha**
Vedoucí práce: **Ing. Petr Hořejší, Ph.D.**

Akademický rok 2016/2017

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jan PRŮCHA**

Osobní číslo: **S14B0221P**

Studijní program: **B2301 Strojní inženýrství**

Studijní obor: **Průmyslové inženýrství a management**

Název tématu: **Plánování výroby pomocí Advanced Planning Scheduling**

Zadávací katedra: **Katedra průmyslového inženýrství a managementu**

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

1. Úvod
2. Analýza současného stavu (APS systémy na trhu)
3. Obecná funkčnost APS systémů
4. Obecné vstupy a výstupy systémů
5. Algoritmy plánování využité v APS
6. Závěr

Rozsah grafických prací: 0 výkresů
Rozsah kvalifikační práce: 30 - 40 stran
Forma zpracování bakalářské práce: tištěná

Seznam odborné literatury:

1. MAUERGAUZ Y. *Advanced Planning and Scheduling in Manufacturing and Supply Chains*. Springer 2016. ISBN 978-3319275215
2. EL-BERISHY N. *An Advanced Production Planning and Scheduling System: for Batch Process Industry*. LAP LAMBERT, Academic Publishing, 2011. ISBN 978-3846558058
3. BASL, J., Velkoborský J. *Přehled českého trhu softwarových nástrojů APS a CRM*. Computerworld 32/2000
4. MAREŠ, J. *Podnikové informační systémy a DP*, e book. Plzeň: SmartMotion, 2012. ISBN 978-80-87539-05-7

Vedoucí bakalářské práce:

Ing. Petr Hořejší, Ph.D.

Katedra průmyslového inženýrství a managementu

Konzultant bakalářské práce:

Doc. Ing. Zdeněk Ulrych, Ph.D.

Katedra průmyslového inženýrství a managementu

Datum zadání bakalářské práce: 19. září 2016

Termín odevzdání bakalářské práce: 2. června 2017



Doc. Ing. Milan Edl, Ph.D.
děkan



Doc. Ing. Michal Šimon, Ph.D.
vedoucí katedry

V Plzni dne 19. září 2016

Prohlášení o autorství

Předkládám tímto k posouzení a obhajobě bakalářskou/diplomovou práci, zpracovanou na závěr studia na Fakultě strojní Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně, s použitím odborné literatury a pramenů, uvedených v seznamu, který je součástí této bakalářské/diplomové práce.

V Plzni dne:

.....
podpis autora

ANOTAČNÍ LIST BAKALÁŘSKÉ PRÁCE

AUTOR	Příjmení Průcha	Jméno Jan	
STUDIJNÍ OBOR	B2301 Strojní inženýrství, Průmyslové inženýrství a management		
VEDOUCÍ PRÁCE	Příjmení (včetně titulů) Ing. Hořejší, Ph.D.	Jméno Petr	
PRACOVIŠTĚ	ZČU - FST - KPV		
DRUH PRÁCE	DIPLOMOVÁ	BAKALÁŘSKÁ	Nehodící se škrtněte
NÁZEV PRÁCE	Plánování výroby pomocí Advanced Planning Scheduling		

FAKULTA	strojní	KATEDRA	KPV	ROK ODEVZD.	2017
----------------	---------	----------------	-----	--------------------	------

POČET STRAN (A4 a ekvivalentů A4)

CELKEM	47	TEXTOVÁ ČÁST	42	GRAFICKÁ ČÁST	5
---------------	----	---------------------	----	----------------------	---

STRUČNÝ POPIS (MAX 10 ŘÁDEK) ZAMĚŘENÍ, TÉMA, CÍL POZNATKY A PŘÍNOSY	Cíle této práce jsou analýza současného stavu APS na českém i zahraničním trhu, fungování APS jako systému i jako softwaru a jeho zhodnocení oproti ostatním plánovacím systémům využívaných v podnicích. Další cíle jsou zaznamenání a vysvětlení obecných vstupů a výstupů APS systému a popis vlastního algoritmu zpracovaného k této práci jako praktickou část.
KLÍČOVÁ SLOVA ZPRAVIDLA JEDNOSLOVNÉ POJMY, KTERÉ VYSTIHUJÍ PODSTATU PRÁCE	APS, řízení výroby, plánování, kapacity, algoritmy, aplikace

SUMMARY OF DIPLOMA (BACHELOR) SHEET

AUTHOR	Surname Průcha	Name Jan	
FIELD OF STUDY	B2301 Mechanical Engineering, Industrial Engineering and Management		
SUPERVISOR	Surname (Inclusive of Degrees) Ing. Hořejší, Ph.D.	Name Petr	
INSTITUTION	ZČU - FST - KPV		
TYPE OF WORK	DIPLOMA	BACHELOR	Delete when not applicable
TITLE OF THE WORK	Production Planning Using Advanced Planning Scheduling		

FACULTY	Mechanical Engineering	DEPARTMENT	Industrial Engineering and Management	SUBMITTED IN	2017
----------------	------------------------	-------------------	---------------------------------------	---------------------	------

NUMBER OF PAGES (A4 and eq. A4)

TOTALLY	47	TEXT PART	42	GRAPHICAL PART	5
----------------	----	------------------	----	-----------------------	---

BRIEF DESCRIPTION TOPIC, GOAL, RESULTS AND CONTRIBUTIONS	Aims of this thesis are the analysis of the current state of APS on Czech and foreign markets, the functioning of APS as a system as well as software and its appreciation compared to other planning systems used in companies. Other aims are to record and explain the general inputs and outputs of the APS system and describe the algorithm, which was created as a practical part of this thesis.
KEY WORDS	APS, production management, planning, capacities, algorithms, applications

Poděkování

Na tomto místě bych rád poděkoval vedoucímu mé bakalářské práce Ing. Petru Hořejšímu, Ph.D., za cenné připomínky a odborné rady při zpracovávání bakalářské práce.

Obsah

1	Úvod.....	11
2	Počátky APS systémů.....	13
2.1	MRP II.....	13
2.1.1	Zpětné plánování dle MRP II.....	13
2.1.2	Dopředné plánování dle MRP II.....	14
2.2	APS jako následník MRP II.....	14
2.2.1	Dopředné plánování APS.....	15
2.2.2	APS s kombinovaným plánováním.....	15
2.2.3	Simulace.....	16
3	Obecná funkčnost APS.....	17
3.1	Rozdělení schopností APS systémů podle funkčních oblastí.....	17
3.1.1	Plánování výroby.....	17
3.1.2	Nákup materiálu.....	17
3.1.3	Obchod.....	17
3.1.4	Management.....	17
3.2	Model řízení podnikových procesů.....	17
3.3	Základní úrovně podnikových procesů.....	18
3.3.1	První úroveň.....	18
3.3.2	Druhá úroveň.....	18
3.3.3	Třetí úroveň.....	18
3.3.4	Čtvrtá úroveň.....	18
3.3.5	Pátá úroveň.....	18
3.4	Koncepty používané v APS.....	19
3.4.1	JIT.....	19
3.4.2	Drum-Buffer-Rope.....	19
3.4.3	TOC.....	20
4	Analýza současného stavu.....	21
4.1	Zavádění APS systémů v českých podnicích.....	21
4.1.1	Nevýhody APS systémů.....	21
4.1.2	Výhody APS systémů.....	22
4.2	APS systémy na trhu.....	23
4.2.1	Simio.....	23
4.2.2	APS Karat.....	24
4.2.3	Addon Production.....	25

4.2.4	Altus Vario	26
4.2.5	Asprova	27
4.2.6	Asprova jako plánovací software	27
4.2.7	Plánování v systému Asprova	28
5	Vstupy a výstupy	30
5.1	Vstupy	30
5.1.1	Databáze	30
5.1.2	Základní vstupy	31
5.2	Výstupy	32
5.2.1	Vedlejší výstupy	32
5.2.2	Simulace	33
6	Algoritmy využívané v APS systémech	34
6.1	ATP	34
6.2	ATTP	34
6.3	CTP	34
6.4	PTP	34
6.5	Frozen Intervals	34
7	Popis praktické části práce	36
7.1	Cíl praktické části práce	36
7.2	Rozhraní formuláře	36
7.2.1	Popis formuláře z uživatelského pohledu	36
7.2.2	Popis formuláře z vývojářského pohledu	37
7.3	Objednávkový list	38
7.4	Vnitřní stavba aplikace	39
7.4.1	Parsování	39
7.4.2	Využití proměnné	40
7.4.3	Dopředné plánování	41
7.5	Nedostatky a zjednodušení dopředného plánování	43
8	Závěr	44
9	Seznam použitých zdrojů	45
10	Seznam Příloh	47

Seznam Obrázků

Obr. 3-1 Vymezené řízení podnikových procesů [1]	19
Obr. 4-1 ERP systémy versus APS systémy [8].....	23
Obr. 4-2 Simio[12]	24
Obr. 4-3 Ganttův diagram v softwaru Karat [16].....	25
Obr. 4-4 Addon Production [17]	26
Obr. 4-5 Úvodní rozhraní Altus Vario [19].....	27
Obr. 7-1 Úvodní zobrazení formuláře	36
Obr. 7-2 Vizualizace jednoduchého plánu	37
Obr. 7-3 Chybné načtení objednávkového listu	38
Obr. 7-4 Ukázka z objednávkového listu	39

Seznam použitých zkratk

APS	Advanced Planning and Scheduling
SCM	Supply Chain Management
ERP	Enterprise Resource Planning
JIT	Just In Time
TOC	Theory of Constraints
MRP	Material Requirement Planning
DBR	Drum Buffer Rope
BOM	Bill of Material
MES	Manufacturing Execution System
MS	Manufacturing Scheduler
DS	Data Server
NLS	Network License Server
ATP	Aviable-to-Promise
AATP	Allocated- Aviable-to-Promise
CTP	Capable-to-Promise
PTP	Profiteble -to-Promise
MPS	Master production schedule
CRP	Capacity requirements planning
DATP (d-ATP)	Dynamic Advanced Planning and Scheduling

1 Úvod

V následujícím bakalářské práci je rozpracována analýza plánování výroby pomocí APS systémů. V práci je rozpracován pohled na dnešní situaci APS na trhu jak v České republice, tak ve světě s detailním nahlédnutím do softwaru Asprova. Dále se práce zabývá obecnou i konkrétní funkčností APS systémů s popisem aplikace, která využívá některé funkce APS.

Na začátku budou vysvětleny základní pojmy této práce. *Advanced Planning and Scheduling (APS)*, česky *Pokročilé plánování a rozhodování* nebo *Plánování do omezených kapacit*, je řešení postupů pro plánování výroby i logistiky s omezenými zdroji. Tedy APS simuluje a hledá řešení pro výrobní plán a bere v potaz i dostupnost materiálu a další zadaná omezení. Dále může poskytovat podnětné informace a propočty při přijímání zakázek.

Řízení výroby v podniku je možné rozřadit na několik vnitřních systémů. Tyto subsystémy mohou být brány z různých hledisek, například z časového hlediska, organizačního hlediska, podle metodiky nebo nástrojů atd. V těchto subsystémech také můžeme najít systém fungující na principu operativního řízení výroby. Operativních řízením výroby se rozumí nepřetržitý cyklus sběru informací, jejich vyhodnocování a aktualizace rozhodování v oblastech:

- Uspokojení zákazníka
- Konkurenční výhody firmy (minimalizace nákladů)
- Vyráběné množství
- Výrobní plán
- Využití kapacit ve výrobním procesu, racionalizace průběhu výrobního procesu
- Logistika (výrobní a nákupní termíny)
- Nákup materiálu, výběr dodavatele
- Stavby zásob, ať už materiálu, rozpracovaných výrobků, nebo hotových výrobků

Na základě tohoto operativního principu se začalo používat systémů pokročilého plánování a rozvrhování, tedy APS.

APS systémy jsou vytvářené hlavně pro podniky, které vyrábějí na zakázku a které mají velkou škálu složitých výrobků, jejichž díly se vyrábějí na stejných pracovištích. APS systémy jsou také vhodně pro podniky, které jsou nuceny často

První systémy pokročilého plánování se začaly vyvíjet v druhé polovině 90. let minulého století, nicméně do České republiky přišly o dekádu později. V podstatě se jedná o nástupce metod MRP (*Material Requirements Planning*) a JIT (*Just in Time*). APS systémy změnily chápání procesu tvorby výrobního plánu. Na výrobní plán se aplikovala umělá inteligence, tedy matematická logika spojená s heuristickým přístupem a tzv. dopředným i zpětným plánováním.

Dá se říct, že APS respektuje MRP a JIT a přebírá jejich požadavky na sebe, tj. zajišťuje přesné a spolehlivé dodávky výrobků k zákazníkovi s důrazem na eliminaci všech neproduktivních činností. Ve většině případů plánování netrvá déle než pár minut a plány nejsou idealizované, ale jsou reálné a dosažitelné (při platných vstupních datech).

Další vývoj APS systémů směřuje k řízení logistiky SCM *Supply Chain Management*, česky *Řízení dodavatelských řetězců*. Tento modul řeší dodávky materiálu, kapacitu dopravovaného množství, čas dopravy. Se získanými informacemi pak pomáhá zracionalizovat výrobní plán.

APS systémy a SCM systémy mají hodně společného, proto jsou také často označovány společnou zkratkou APS/SCM. Rozdíl mezi SCM a APS spočívá v tom, že SCM řeší plánování venkovní, tedy v rámci dodavatelského řetězce a APS řeší plánování uvnitř podniku.

Pro přehlednost jsou uvedeny rozdíly mezi plánováním klasickým a plánováním s použitím APS.

Klasické plánování

- Lokální racionalizace výrobního plánu
- Sekvenční plánování
- Velmi omezené možnosti v oblasti zakomponování omezení do výrobního plánu
- Sledování změn jen v jednom směru a vícenásobné opakování
- Omezené zobrazení reality produkce

Plánování s APS

- Celková racionalizace
- Simultánní plánování
- Plánování s omezeními
- Sledování vícefaktorových změn a vysoká rychlost
- Přesné zobrazení dodavatelského řetězce [1]

Největší problém APS systémů je fakt, že je velmi těžké využívat APS správně. APS vyžaduje naprosto korektní a přesné data. Pokud je nedostane, výsledný plán může být chybný.

Cíle této práce jsou analýza současného stavu APS na českém i zahraničním trhu, fungování APS jako systému i jako softwaru a jeho zhodnocení oproti ostatním plánovacím systémům využívaných v podnicích. Další cíle jsou zaznamenání a vysvětlení obecných vstupů a výstupů APS systému, vysvětlení fungování základních algoritmů používaných v APS a algoritmus vlastní tvorby na téma APS.

2 Počátky APS systémů

Pokročilé plánování a rozhodování bylo navrženo na základě manažerského principu, který prosazuje využívání výrobní kapacity a materiálu pro uspokojení zákazníka. Tedy například pokud má podnik rozpracovanou zakázku s termínem až ke konci měsíce a jiný zákazník přijde s objednávkou, kterou by chtěl do konce týdne, plánovač podniku se bude snažit vyhovět oběma zákazníkům. To samozřejmě zcela změní výrobní plány výrobních pracovišť. A to je případ, kdy jednoduché plánovací metody, nejčastěji prováděné pomocí programu Microsoft Excel, selhávají a nejsou schopny vyprodukovat plán výroby, který by vyhovoval nynějšímu, dynamicky se měnícímu stavu objednávek.

Ve zkratce se dá říci, že APS systémy jsou vhodné tam, kde jednodušší metody plánování nezvládají vícekritériální rozhodování.

2.1 MRP II

Pro lepší pochopení APS systému nejdřív bude vysvětlena základní funkčnost kapacitního plánování v MRP II, předchůdce APS.

Nejprve se vytváří hrubý plán zvaný MPS, tedy *Master production schedule*. Hrubý plán se sestává z finálních položek kusovníku, tedy většinou hotových výrobků, který chce firma vyrobit a to v střednědobém časovém horizontu. Tento hrubý plán zanedbává rozpracovanost výroby i jakékoli kapacity strojů a pracovišť.

Hrubý plán se používá jako základní kostra, od které se odvíjí plán hlavní, který se označuje jako MRP, *Manufacturing Resources Plan*. V tomto plánu se řeší jednotlivé úrovně kusovníku, množství materiálu, které tyto úrovně kusovníků budou potřebovat a čas potřebný pro započítání práce na jednotlivých položkách kusovníku. Pokud je položka nakupovaná, neřeší se čas započítání práce, ale doba nákupu. Hlavní plán zanedbává možnost nedostupnosti materiálu.

Jakmile se vytvoří hlavní plán, začne samotné kapacitní plánování označované jako CRP z anglického *Capacity requirements planning*. CRP zkoumá možnost, kdy některý z výrobních zdrojů je v důsledku hlavního plánu přetížen. Výrobními zdroji jsou myšleny lidské zdroje, strojní kapacity atd. Pokud se ukáže, že tato možnost je někde reálná, tedy že při současném hlavním plánu bude jeden nebo více výrobních zdrojů přetížen, musí plánovač upravit hlavní (a případně i hrubý) plán. Ovšem zasahování do plánů je složité, pokud cílem zasahování je odstranit přetíženost daného zdroje a zároveň nepřetížit žádný jiný zdroj. Toto zasahování je tím víc složitější, čím má podnik, kde se plánování odehrává, větší počet výrobních zdrojů, jelikož se musí zkoordinovat všechny zdroje, kterých se tento zásah týká.

2.1.1 Zpětné plánování dle MRP II

Nespornou výhodou zpětného plánování je jeho jednoduchost a necitlivost co se týče přesnosti dat a tedy i levná cena provozování aplikace založené na tomto principu. Nicméně nevýhody tohoto principu plánování jsou obrovské. Jedná se o zanedbání dostupnosti materiálu i omezenosti kapacit (ty se řeší až po naplánování). Dále je zde při plánování naprostá absence vazeb mezi jednotlivými zakázkami. Použití zpětného plánování MRP a MRP II však prakticky znemožňuje i při jednoduché výrobě dosáhnout realistického hlavního plánu i po zpětných zásazích plánovače. Čím větší bude množství objednávek, typů výrobků a vytíženost pracovišť, tím hůř půjde vytvořit použitelný hlavní plán, i když se do něj bude zpětně zasahovat.

Je tedy zřejmé, že použití MRP II ve své původní podobě nemůže zajistit firmě v dnešní době funkční a splnitelný plán, který může určit datum ukončení jednotlivých zakázek. Tento důvod vedl k tomu, že se zavedlo dopředné plánování s uvažováním omezených kapacit.

2.1.2 Dopředné plánování dle MRP II

Dopředné plánování dle MRP II, někdy nazývané Post MRP II, je odpověď na nedokonalost plánování zpětného. Na rozdíl od zpětného plánování, plánování dopředné plánuje do omezených zdrojů. Vstupní data jsou stejně jako v zpětném plánování hrubý a hlavní plán.

Hlavní rozdíl mezi oběma typy plánování je ten, že dopředné plánování má začátek časové osy v současnosti. Druhá odlišnost je v plánování kapacitním. V dopředném plánování se pracoviště ideálně zatěžují na jejich maximální kapacitní výkon. Tedy v praxi práce je naplánována na pracoviště hned, jak je k dispozici materiál z předchozí operace. Tento postup je vhodný pro silně zatížené pracoviště a naopak nevýhodný pro pracoviště s malým využitím. Pracoviště s malým využitím by při použití tohoto principu zbytečně zpracovávala materiál příliš brzy a rozpracovanost výroby by se odrazila na zvýšených nákladech.

Tento princip plánování tedy umožňuje plánovat se skutečnými kapacitami zdrojů a umožňuje případnou optimalizaci jednotlivých zdrojů. Nevýhodou je značná pracnost plánování v případě, že některé pracoviště není schopné zvládnout všechnu práci a musí se část práce přesunout na pozdější dobu. Dále tento princip vyžaduje opravdu přesná data, což může pro některé podniky být problém.

Dopředné plánování dle MRP II, nevyužívá metodu TOC a proto se operace na kritických pracovištích zpožďují. Díky dopřednému plánování se sice podnik vyhne přetížení kritických pracovišť, nicméně při přesouvání práce na pozdější dobu u kritických zdrojů nedochází ke stejnému přesunutí dodání materiálu. Takže se materiál objedná zbytečně brzy, čímž se zvýší náklady na skladování. Druhá možnost je, že se při posunutí práce na pozdější dobu budou hromadit rozpracované díly potřebné na tuto práci, čímž se zase zvyšuje rozpracovanost. Jako další problém lze uvést nepřesnost při více produktové výrobě. V případě, kdy jedno pracoviště má podle plánu provést operace z různých zakázek, Post MRP II požadavky všech zakázek sečte dohromady, místo aby s nimi nějak pracoval dále. Výsledek je přetížení daného pracoviště.

Celkově tedy nadstavba MRP II, Post MRP II, příliš nezlepšila plánování výroby jako takové. Zpoždění zakázek se nepodařilo odstranit, kvůli přesouvání výroby do budoucna se výroba roztříštila a propustnost výroby se stejně nezvýšila. Post MRP II nedala moc možností více produktovým podnikům bez linkových pracovišť. Výrobní plán tedy opět není reálný a slouží jen jako určitá osnova výroby. [6]

2.2 APS jako následník MRP II

Klasické MRP II systémy, tedy postupně procházejí jednotlivé technologické operace a přiřazují jim potřebný materiál a výrobní kapacitu. MRP II ale nejsou nepřizpůsobené k rychlým změnám v požadavcích na výrobní zdroje a k počítání s reálnou kapacitou materiálu. APS systémy jsou vyvíjeny s důrazem právě na plánování s požadavky na materiál, pravidly a omezeními podniku a dostupnou kapacitou výrobních pracovišť. Hlavní výhodou APS systémů je použití optimalizačních metod pro tvorbu výrobního plánu tak, aby zohlednil aktuální stav priorit v plánu. V praxi tedy APS zkrátí plánovací čas na minimum a umožní mu více se zaměřit na úzká místa a efektivitu plánování. [2]

Rozdíl mezi MRP II a APS při vytváření výrobního plánu se dá zjednodušit do 5 bodů, které v MRP II systémech nenajdete:

- APS systémy pracují s dostupností veškerých výrobních zdrojů
- APS systémy podporují metodu TOC a zohledňují kritické výrobní zdroje
- Výrobní zakázky jsou plánovány všechny najednou a se zohledněním na jejich priority
- Výrobní dávka není fixní, ale je tvořena dynamicky podle aktuální potřeby
- Průběžná doba výroby není fixní, ale je výsledkem výrobního plánu
- APS systémy umožňují plánování alternativních postupů

Důvody ke vzniku a rozdíly s MRP II jsou tedy známy, nicméně APS systémy prošly od doby svého vzniku určitým vývojem, který je dále nastíněn.

2.2.1 Dopředné plánování APS

Nejprve byly APS systémy brány jako aplikace s dopředným plánováním, které plánovaly s veškerými druhy omezení, které byly v daném podniku. Při plánování se počítalo s prioritami jednotlivých zakázek, s dostupností materiálu a kapacity jednotlivých strojů, lidí i různých přípravků. Aplikace při plánování kontrolovala dostupnost všech zdrojů, které budou při každé operaci potřeba. Pokud některý zdroj dostupný nebyl, operace nebyla naplánována.

Tím se velice zvýšila propustnost systému, termíny dodávek byly mnohem přesnější a plány byly reálně proveditelné. Nicméně problémy jako roztříštěnost výroby a velké skladové náklady přetrvávaly. Chybí zde element ze zpětného plánování, který by zajišťoval synchronizaci nákupů a výdejů materiálu.

2.2.2 APS s kombinovaným plánováním

Po zjištění problému na principu dopředného plánování APS byla vyvinuta kombinace dopředného pokročilého plánování a starého zpětného plánování dle MRP II. Tvorba plánu zde probíhá odlišně od předešlých principů plánování.

Nejprve přijde ke slovu plánování dopředné, které seřadí operace podle kritických zdrojů, všech omezujících faktorů a priorit zakázek. Případně bere v potaz i plánovačem nastavené faktory jako například velikost bufferů, velikost výrobních a dodacích dávek nebo maximální výše rozpracovanosti. Tím se zajistí propustnost systému.

Zpětné plánování pak působí na nekritické pracoviště, které taktuje podle pracovišť kritických, aby se zajistila synchronizace operací. V tomto plánovacím kroku se využívají metody JIT, TOC a Drum-Buffer-Rope. Každá operace je plánována právě včas tak, aby výrobek mohl hned jít na následující kritické pracoviště.

Výsledný plán kombinace dopředného a zpětného plánování není roztříštěný, ale je synchronizovaný (kritická místa taktují místa nekritická). Náklady i rozpracovanost jsou sniženy o několik desítek procent. Plán jako takový je skutečně realizovatelný. [6]

Metoda APS byla vyvinuta primárně pro výrobu na objednávku, kde je poptávka časově proměnná, pro podniky s velkým počtem výrobků, které se dělali na stejných pracovištích a pro složité výrobky s mnoha komponenty, nicméně pro schopnost reagovat na častou změnu výrobního plánu se uplatní ve většině výrobních prostředí.

APS systémy jsou v podnicích provozovány buď samostatně, jako nástroj pro plánování výroby, nebo jsou použity jako moduly nějakého ERP systému.

2.2.3 Simulace

APS systémy jsou ceněny z velké části také pro svojí schopnost vytvářet simulace vývoje výroby na reálných i nereálných datech, bez nebezpečí, že by simulace nějak ovlivnila stávající výrobu a výrobní plán. Simulace je popsána níže v kapitole „Vstupy a výstupy“. [4]

3 Obecná funkčnost APS

V následující kapitole jsou rozebírány obecné funkční prvky APS.

3.1 Rozdělení schopností APS systémů podle funkčních oblastí

3.1.1 Plánování výroby

Plánování výroby je jeden ze základních kamenů APS systémů. Zde jsou ve stručnosti zopakované funkce APS ve vztahu k plánování výroby: Vytváří plán výroby s ohledem na omezení jako kapacita materiálu, kapacita pracoviště a termín zakázky a expedice. Vypracovává přesný plán výroby pro jednotlivé pracoviště a jednotlivé stroje. Vyhledává úzká místa a plán podle úzkých míst optimalizuje, čímž snižuje čas výroby a potřebné množství kapacity pracovišť. Rychlost výpočtů při změně údajů bývá několikrát rychlejší než u průměrných ERP systémů.

3.1.2 Nákup materiálu

APS systémy pracují s principem JIT (*Just in time*), takže pomáhají snížit zásoby a naopak zlepšit a zefektivnit dovoz materiálu při zachování stejného objemu výroby. Systémy většinou samy vytváří požadavky na nákup, tedy kdy se má jaký materiál nakoupit. Po integrování APS systému sníží podniky v průměru o 25% objem zásob.[5]

3.1.3 Obchod

Nežádka se stává, že obchodní oddělení přijme prakticky nesplnitelnou zakázku. Při integrování APS do podniku je možné u každé zakázky ověřit, zda je její termín reálný, respektive, zda se vyplatí ji přijmout i když se termín dodání nedodrží. Lze plánovat nebo rezervovat kapacity pro obchodní výhledy a zatím nepotvrzené zakázky, pokud je tak v zájmu podniku. Díky propojení jednotlivých oblastí podniku je obchodní oddělení a management informován o zakázkách, které mají ohrožené splnění termínu dodání.

3.1.4 Management

Díky racionalizaci výrobního plánu je zvýšený objem výroby a zkrácená průběžná doba výroby. APS systém obecně zvyšuje přesnost dodání zakázek, čímž se zvyšuje dodavatelská spolehlivost. APS systémy snižují zásoby materiálu a zásoby rozpracovaných výrobků. APS softwary také poskytují podklady pro strategická rozhodnutí k navýšení kapacit strojů.[5]

3.2 Model řízení podnikových procesů

V dnešní době je nezbytnou podmínkou každého podniku zajištění kvalitního podnikového informačního systému s odpovídající informační technologií.

Model řízení podnikových procesů popisuje integraci řízení podnikových procesů a řízení všech oblastí využívajících podnikové zdroje. Model je tvořen pěti základními úrovněmi řízení podnikových procesů a třemi oblastmi podnikových zdrojů. Obě části modelu jsou provázané úzkými vazbami.

3.3 Základní úrovně podnikových procesů

3.3.1 První úroveň

První úroveň je v podstatě strategické určení řízení podnikových procesů. Jedná se o rozhodnutí na základě cílů a vize podniku a politikou podniku napříč všemi spektry.

Výstupy první úrovně jsou strategická rozhodnutí, jako stanovení podnikových cílů, rozhodnutí o způsobu získávání lidských, finančních i informačních zdrojů a definování úspěšnosti podniku.

3.3.2 Druhá úroveň

Druhá úroveň, nebo-li úroveň rozlišovací, je určena k definování podnikových procesů na základě definovaných produktů nebo služeb z úrovně první. Účel tohoto definování je stanovit charakter daného produktu či služeb, ten je pak základem pro návrh podnikových procesů. Na druhé úrovni jsou tvořeny procesy na obecné úrovni, tedy něco jako hrubý model řízení podnikových procesů, který je pak základem pro vznik detailnějšího plánování v třetí úrovni. Hlavní výstupy druhé úrovně jsou:

- Návrh hlavních procesů, například obchodních procesů
- Vymezení procesů podpůrných, například řízení lidských zdrojů, marketingových procesů
- Popis každého procesu od zahájení procesů přes stanovení funkčních míst, odpovědných za jednotlivé činnosti, po návaznost procesů
- Výpočty spotřeby podnikových zdrojů a potřebného času
- Stanovení kritérií pro zpětnou vazbu

Z výčtu výstupu lze snadno odvodit, že zde dochází k jednomu z nejdůležitějších kroků při řízení podnikových procesů. Právě zde se ke slovu dostávají softwary pro vytváření a optimalizaci podnikových procesů jako APS systémy.

3.3.3 Třetí úroveň

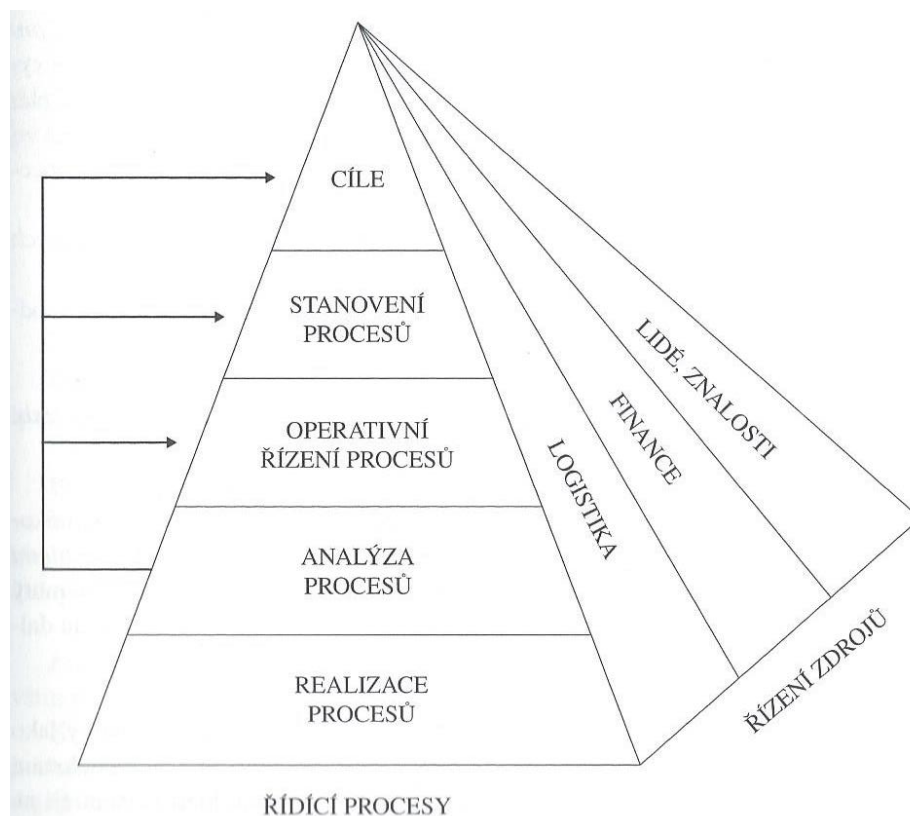
Třetí úroveň je zaměřena na operativní řízení podnikových procesů. Jde zde především o kontinuální rozhodování v řízení podnikových procesů a kapacit. Výsledkem třetí úrovně by měl být detailní operativní plán řízení procesů a zdrojů podniku se zřetelem na požadavky zákazníků. Jako další výstup této úrovně by mělo být stanovení kontrolních zásahů do řízení procesů v důsledku aktuálních změn v zákaznických objednávkách.

3.3.4 Čtvrtá úroveň

Tato úroveň je zaměřena na monitorování průběhu podnikových procesů. Monitorování průběhu procesů je důležité kvůli analýze a hodnocení pro zpětnou vazbu a pro možnost rychlých reakcí při neplánovaných incidentech

3.3.5 Pátá úroveň

Poslední úroveň je zaměřena na vlastní realizaci podnikových procesů, které jsou nadefinované a optimalizované z předchozích úrovní. Činnosti probíhající na této úrovni mají výkonný charakter. Výstupem páté úrovně jsou hotové produkty nebo služby úspěšně doručené zákazníkovi.[6]



Obr. 3-1 Vymezené řízení podnikových procesů [1]

3.4 Koncepty používané v APS

APS systémy využívají hned několik přístupů a metod k plánování výroby.

3.4.1 JIT

JIT je zkratka pro *Just in time* (český překlad *Právě včas* se prakticky nepoužívá). Název této metody je výstižný, neboť se opravdu jedná o sestavení dodávek materiálu tak, aby byly dopraveny v potřebnou dobu a v potřebném množství. Opoždění anebo nedostatek materiálu je stejně nežádoucí jako příliš včasná dodávka, nebo dodávka příliš velká, jelikož materiál na skladě k sobě váže kapitál.

Tato metoda předpokládá spolehlivou spolupráci ze strany dodavatelů i zákazníků ve smyslu včasných dodávek od dodavatele a okamžitým odběrem zákazníka. Pomocí této metody má podnik minimální zásoby, které kompenzují časté dodávky materiálu v menším objemu. Tato metoda byla vyvinuta v Japonsku, kde se podle ní řídí velké procento podniků. Bohužel, tuzemské firmy zpravidla nedosahují takových úspor zásob jako v Japonsku kvůli horší spolupráci s dodavateli a zákazníky.[4]

3.4.2 Drum-Buffer-Rope

Tento koncept, občas jmenován jen jako *DBR*, obsahuje pozitivní principy MRP II a je aplikován s technologií JIT.

Drum, česky *buben* představuje rozvrhování činností omezeného zdroje, tedy kritické místo, kapacitní omezení podniku. Dále *buben* udává rytmus práce celé výroby. V praxi se tedy jedná o plán výroby, který zohledňuje kritická místa výroby.

Buffer, česky *zásobník*, má funkci ochrany před zpožděním termínu. Jedná se tedy o jakési časové a kapacitní zajištění před nečekanými výpadky a problémy během výroby. Tento nástroj ochraňuje propustnost provozu, ale při jeho předimenzování může degradovat efekty APS.

Rope, česky *lano* představuje nástroj pro synchronizaci všech nekritických operací či míst do rytmu *bubnu*, tedy kritického místa. Jedná se o nástroj zajišťující stabilitu plánu a dodavatelskou spolehlivost při minimálních nákladech ve věci zásob.[8]

3.4.3 TOC

Teorie omezení, anglicky *Theory of Constraint*, je koncept plánování založeným na principu, že schopnost plánování je omezována jedinou překážkou – omezeným zdrojem neboli úzkým místem. Teorie omezení funguje tak, že všechny operace před úzkým místem pracují v taktu udávaným právě úzkým místem. Operace následující po úzkém místě jsou pak tlačeny k takové rychlosti výroby, jakou dovolí průtok materiálu omezeným zdrojem. Omezený zdroj může být fyzické i nefyzické povahy, například stroj i procedurální omezení. Postup v souladu s konceptem TOC je následující:

1. Najít omezení systému
2. Rozhodnout, jak nejúčinněji využít omezení
3. Podřídít vše ostatní tomuto rozhodnutí
4. Rozšířit, případně jinak posílit, existující omezení
5. Vrátit se k bodu jedna a hledat nové omezení.

Teorie omezení se zaměřuje na nejslabší články, protože zesílení jiného článku systém neposílí, dokud nejslabší článek zůstane nezměněn. Nejčastěji se tento princip přirovnává k článkům řetězu.[4]

4 Analýza současného stavu

4.1 Zavádění APS systémů v českých podnicích

Na začátku této kapitoly je třeba připomenout, že ne vždy má zavedení pokročilého plánování do podniku smysl. APS se obecně vyplatí spíše u velkých firem, ačkoli existují i řešení pro firmy malé.

Když už se podnik rozhodne pro zavedení nějakého APS systému, není vůbec snadné vybrat správné řešení. Pro zákazníka, který se v APS neorientuje příliš dobře, se může zdát, že bude problém rozlišit výhody a nevýhody jednotlivých APS řešení. A ani orientace v APS softwarech není všespásná. Zákazník, případně jím pověřený pracovník, musí být schopen posoudit, zda on a jeho spolupracovníci mají dostatečný nadhled nad interními procesy podniku a tím jsou schopni upravit procesy podle nového APS systému sami, nebo radši svěří implementaci do rukou odborníků, kteří nabízejí k APS řešení i poradenství.

Implementace APS systému, nebo vůbec jakéhokoli informačního systému představuje pro podnik velkou změnu, která se netýká jen nové technologie, ale i podnikových procesů, podnikové kultury. A každá změna představuje pro zaměstnance negativní prvek, protože u každé změny dochází na začátku k nedostatku času. Pro zaměstnance vznikají jiné postupy a jiné úkoly, které nejsou „zaběhnuté“ v praxi. K těmto novotám samozřejmě patří i každodenní práce zaměstnance, takže lidé přestávají stíhat a začínají být s informačním systémem nespokojeni. Bývá proto vhodné lidem přicházejícím do styku s novým systémem dočasně ulevit od každodenní práce, nebo je nějak jinak kompenzovat. Management by měl tyto zaměstnance kontrolovat, informovat a komunikovat s nimi, aby se zaměstnancům systém zapracoval. V opačném případě se může stát, že zaměstnanci začnou opět pracovat „postaru“ a celá implementace informačního systému přijde vniveč.

4.1.1 Nevýhody APS systémů

Přestože jsou metody APS známé už poměrně dlouho, nejsou v České republice příliš rozšířené. Zde jsou důvody „neoblíbenosti“ systému APS i přes jejich nesporné výhody.

Využití přístupů APS u tuzemského průmyslu je kvůli několika problémům nebo nevýhodám nabízených softwarů. Jako první důvod lze uvést přílišné nároky na uživatele. Programy jsou velmi komplexní a kladou na uživatele nároky, které uživatel často nechce, nebo nemůže akceptovat. Ať už kvůli nákladům s nimi spojeným, nebo kvůli náročnosti a zátěži zaměstnanců. Podniky nejsou schopné udržovat svá data na určité úrovni a kvalitě a zaměstnanci obvykle změnu zaběhlého systému nevitají. Někteří prostě nemají rádi změny u něčeho, co zatím „tak nějak“ funguje, další mohou mít obavy se zvládnutím ovládnutí svého sektoru v systému a koneckonců někteří se mohou oprávněně obávat o své místo v podniku, které by mohlo být zavedením softwaru zrušeno.

I kvůli zaměstnanecké nevoli je naprosto nezbytná podpora vrcholového managementu, který musí své zaměstnance donutit akceptovat změnu a opravdu se věnovat školením do nového systému. Dále musí být management připraven prosadit určité změny v procesech podniku, které zavedení nového systému nutně provází. [7]

Další problém bývá nevyhovující datová základna. Tedy pro použití nějakého APS softwaru by byla třeba nová datová základna, která by rapidně zvedla cenu zavedení APS přístupu.

Často očekává vedení podniku od APS přístupu až přehnané schopnosti řízení podniku. Některé podniky předpokládají, že APS software vyřeší jejich potíže s plánováním a řízením zdrojů. Po bližším seznámení samozřejmě přichází zklamání. Déle jsou podniky velmi nepřístupné k nutným organizačním změnám pro zavedení APS systémů, které jsou nezbytnou součástí pro integrování APS, ne-li přímo jedním z hlavních bodů.

Řada podniků o zavedení přístupu APS a jemu podobným nestojí, protože preferují improvizování při chodu výroby a v tomto prostředí se APS integruje velmi obtížně.

APS plánuje kapacity až do toho nejmenšího detailu. V reálném procesu však splnit tento detail je často nereálné. Život přináší změny jak na straně požadavků, tak i na straně kapacit. Optimalizovaný plán někdy druhý den, někdy i za hodinu přestává platit. To vyžaduje neustálé přepočítávání plánu v reálném čase se změněnými podmínkami, tak aby řídicí pracovníci v každém okamžiku věděli o nejbližších úkolech, hrozbě posunu termínu výrobních zakázek, dynamicky se měnících úzkých místech. APS proto často slouží pouze jako agregovaná kontrola, zda lze požadavky v daném čase a s danými zdroji splnit. Plán tak slouží často jako neustále se měnící zásobník práce jednotlivých výrobních zdrojů.

APS systém vyžaduje úzké propojení na ostatní oblasti podnikového řízení: na technickou přípravu výroby, řízení kapacit personálních, materiálových, finančních a výrobně technických zdrojů. APS se neobejde bez sledování a vykazování výroby a bez permanentních změn disponibilních výrobních zdrojů. Často je vyžadováno napojení na automatizační systémy, MES systémy (*Manufacturing Execution System*). Bez propojení na tyto oblasti podniku nelze plánování úspěšně využívat.

Podle mnoha názorů, je nasazení APS poměrně složité, vyžaduje mnoho složitého nastavení, a to je označováno za hlavní důvod sporadického úspěchu.[9]

4.1.2 Výhody APS systémů

Jako jedna z největších výhod integrování APS systému je úspora času při vlastním plánovacím procesu. Zejména u velkých podniků, kde probíhá tisíce operací za den, se plánování zjednoduší několika násobně. Zjednodušení plánování se pak promítne na možnosti mít každý den aktuální podobu plánu s nově nastalými změnami. Další důsledek je možnost změny priority mezi zakázkami či doplňování nově přijaté zakázky. Je možné ještě před potvrzením termínu zakázky a jejím zaplánováním nasimulovat, co která zakázka udělá s kapacitami. Pracovníci se také lépe orientují v pracovních příkazech a mají lepší přehled, jaký materiál kam vstupuje a co je čím vykryto. Jednoznačně se zoptimalizovala fronta práce na zdrojích výroby.

Bez použití plánovacích nástrojů APS je kapacitní plánování velice náročné, většinou plně zaměstnává personální obsazení oddělení plánování a při jakékoli změně (např. porucha stroje) je následné přeplánování pracné. Zavedením systému pokročilého plánování se situace rapidně změní. Časově náročné operace provede samo APS a hlavní úloha oddělení plánování spočívá v hledání možných scénářů, jak vyřešit zakázky, které APS identifikovalo jako termínově nerealizovatelné.[10]

APS systém jsou nejčastěji porovnávány s masově rozšířenými ERP systémy (*Enterprise Resource Planning*, česky *Plánování podnikových zdrojů*). Zde je tedy určité porovnání mezi nimi.

Oblast	ERP systémy	APS systémy
Plánování	Plánování do neomezených kapacit	Reálné plány zahrnující omezené kapacity
	Cíl: proveditelné plány	Cíl: optimální plány
	Tlačný systém	Tažný i tlačný systém
Oblast řízení	Řízení výroby	Řízení celého výrobního řetězce
Hlavní oblasti zaměření	Transakční systém: finance, controlling, výroba	Plánování poptávky, výroby, logistiky, dodavatelského řetězce
Tok informací	Shora dolů	Oběma směry
Schopnost simulací	Nízká až žádná	Vysoká
Schopnost optimalizace nákladů, zisku ceny	Není k dispozici	Vysoká
Doba výroby	Fixní	Flexibilní
Postupné plánování	Není k dispozici	K dispozici
Rychlost plánování a přeplánování	Nízká, obvykle jednou týdně až měsíčně	Vysoká, možné po každé změně
Zpracování dat	Transakční systém	Analytické zpracování dat
Uchování dat	Databáze	Memory-resident

Obr. 4-1 ERP systémy versus APS systémy [8]

Přestože APS může fungovat jako samostatný systém, slušelo by se podotknout, že velká většina APS systémů je používána jako nadstavba ERP systémů.

Návratnost integrace APS systémů je 6 až 12 měsíců.[25]

4.2 APS systémy na trhu

V dnešní době, kdy se systémy APS začínají prosazovat mezi běžnými programy typu ERP, se mnoho firem snaží vyvinout vlastní APS program a zaujmout s ním postavení na trhu.

Rozhodování o tom, jaký typ simulačního softwaru vybrat pro podnikání může být velmi obtížné rozhodnutí. Pro určité nastínění situace na trhu zde budou představeny některé z možných APS systémů. Jeden z nich, Asprova, bude analyzován podrobněji.

4.2.1 Simio

Program Simio od stejnojmenné firmy umožňuje na rozdíl od ostatních APS programů odstraňovat zbytečná rizika ve výrobním programu díky analyzování jejich patentovaným algoritmem *Risk Based Planning and Scheduling*. Kombinace simulace a analýzy tak zlepšuje výkon podniku a obchodní tržby díky racionalizaci využívání kritických zdrojů a vyhodnocování rizika provozních rozhodnutí.[12]

4.2.1.1 Fungování programu

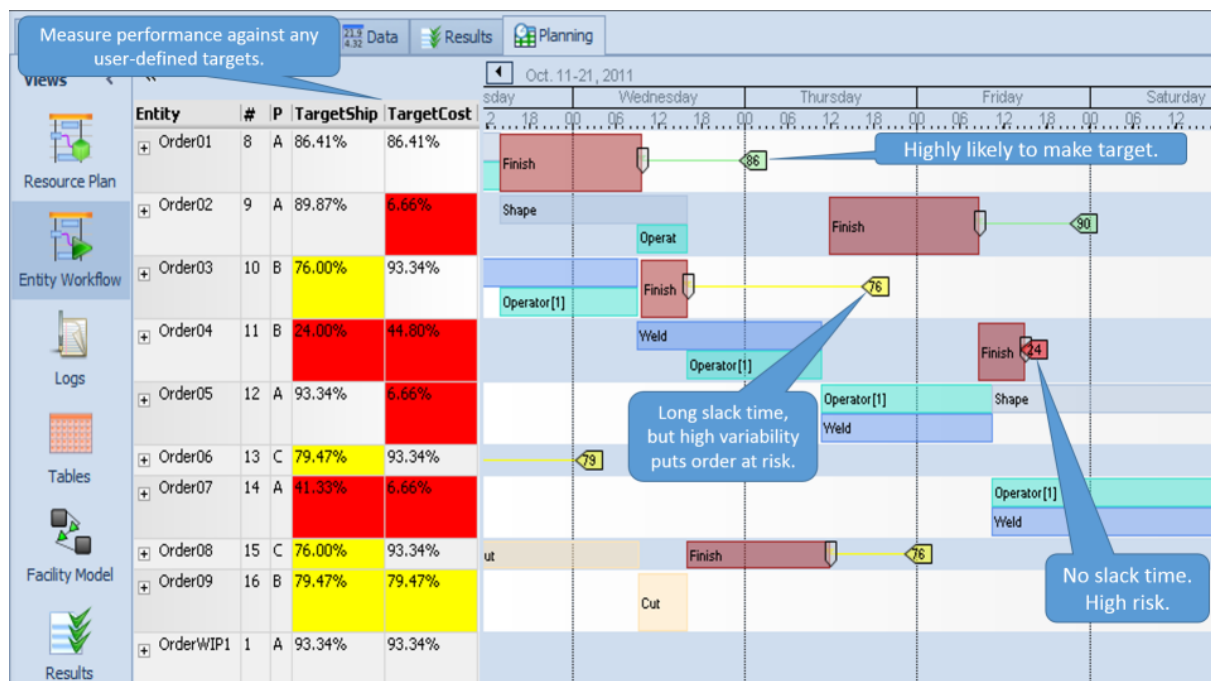
Program vytvoří podrobný rozvrh pro podnik v dokonalém prostředí, tedy prostředí kdy výroba trvá vždy stejně, materiál vždy dorazí včas a stroje se nerozbíjí. Pak se takovýto model zkopíruje do modelu aktivovanými variacemi poruch, zpoždění atd. To umožňuje provádět pravděpodobnostní analýzu, která odhalí rizika rozvrhu. Výstupem tohoto programu je pravděpodobnostní splnění stanovených cílů.

Software umožňuje 2D i 3D animace pro lepší znázornění řešení. Systém také používá cloudová úložiště pro lepší dostupnost dat.[13]

4.2.1.2 Uživatelské rozhraní – výhody a nevýhody

Software dokáže zpracovat i velmi složité systémy. Má více způsobů jak implementovat spouštěcí mechanismy, díky čemuž je uživatelsky přístupnější. [14] Více způsobů vkládání a

spouštění simulací ale na začátku používání softwaru působí nepřehledně. Firma Simio tento problém vyřešila tak, že po zakoupení licence je k dispozici online Simio pracovník, pro uvedení do programu. Simio také připravila průvodní videa, která značně usnadňují výuku ovládání. Díky tomu je Simio vhodný nejen pro pokročilé uživatele plánovacích softwarů ale i pro začátečníky.[32]

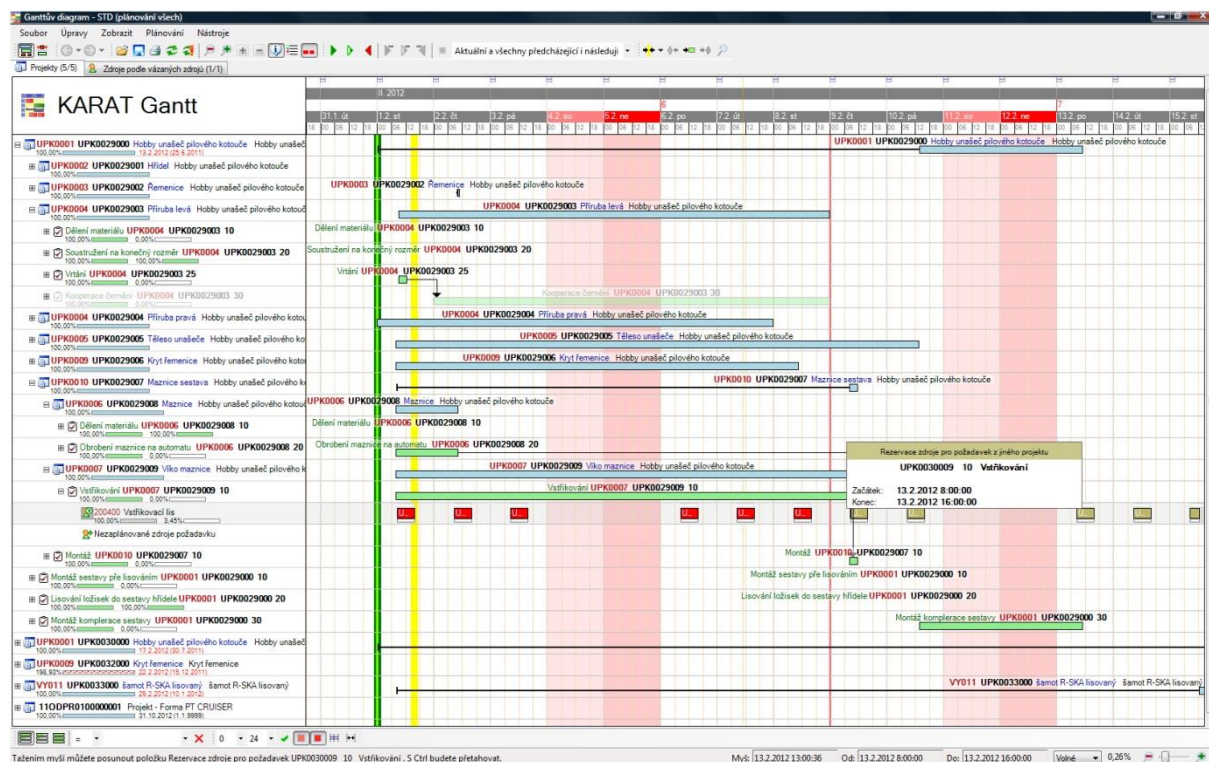


Obr. 4-2 Simio[12]

4.2.2 APS Karat

Systém APS Karat od firmy Karat Software a.s., která se prosadila na českém a slovenském trhu hlavně se svým produktem ERP Karat, je na trhu již od roku 1996. Tento software je zaměřen hlavně na střední a malé podniky, i když je použitelný i pro velké podniky. APS Karat láká zákazníky na klasické výhody APS, tedy snížení průběžné doby výroby, podstatné zkrácení doby dodávek k zákazníkům, optimalizací skladových zásob, efektivnější využití zdrojů firmy a celkovou redukci nákladů.

A také na vylepšený Ganttův diagram, který zajišťuje vizualizaci plánovaných požadavků z hlediska kapacit a jejich zdrojů. [14] APS Karat obsahuje mnohé možnosti plánování. Kromě klasických (strategického, operativního a taktického plánování) i plánování ATP (Avialbe-to-Promise), AATP (Allocated- Avialbe-to-Promise), CTP (Capable-to-Promise), PTP (Profiteble -to-Promise). Průměrná doba implementace u podniku střední velikosti je 6 měsíců.[15]



Obr. 4-3 Ganttův diagram v softwaru Karat [16]

4.2.3 Addon Production

Addon Production byl vyvinut v roce 2006 firmou GTSysytems2 jako rozšíření softwaru SAP® Business One firmy SAP. Tento nadstavbový modul byl vytvořen pro optimalizování řízení výroby v malých podnicích. Tento software je dostupný v českém jazyce. Jedná se o přehledný a na APS softwaru jednoduchý program ideální pro začátečníky v používání těchto softwarů.[17] Daň za jednoduchost je poněkud omezený výběr funkcí. Tento program optimalizuje dle úzkých míst, plánování zde je možné pouze operativně a obsahuje pouze ATP algoritmus. Průměrná doba implementace u podniku střední velikosti jsou 2 měsíce.[18] Mezi zákazníky patří například ArcelorMittal Ostrava a.s. nebo FERRIT s.r.o.[17]

Definovanie technologického postupu

Číslo postupu: 532-1-0013 Typ technológie: Diskrétna

Poznámka: _____

Op	Stredisko	Pracovisko	Názov pracoviska	Názov operácie	Tarif	Kuso...	Kusový čas pra...
10	30	0942301	Ustavovanie	zámočnícke práce	542	3,000	3,000
20	30	1271201	K320 - Zvárací agre	zváranie	542	4,700	4,700
30	50	0931301	Rovnanie - Nitárstv	rovnanie	442	7,000	7,000
40	30	0986301	Kontrola	kontrola	351	0,000	0,000
50	20	0513401	FA3AH - Frézka ho	frézovanie	442	1,600	1,600
60	20	0942101	Zámočnícke práce	zámočnícke práce	352	1,500	1,500
70	20	0941201	Rysovacie práce	rysovanie	442	1,000	1,000
80	20	0464101	VR2 - Vrtáčka radiál	vrtanie	442	5,000	5,000
90	20	0942101	Zámočnícke práce	zámočnícke práce	442	0,700	0,700
100	20	0986301	Kontrola	kontrola	351	0,000	0,000
110	30	0942301	Ustavovanie	zámočnícke práce	542	7,000	7,000
120	30	1271201	K320 - Zvárací agre	zváranie	542	7,700	7,700

Op: 30
Stredisko: 30
Pracovisko: 0942301
Názov pracoviska: Ustavovanie
Názov operácie: zámočnícke práce
Dávka op.: 60,000
Poč. pracovníkov: 1
Poč. strojov: 1
Medzioperačný čas: 0,000 Min.
Kusový čas stroja: 3,000 Min.
Prípravný čas stroja: 10,000 Min.
Kusový čas pracovníka: 3,000 Min.
Prípravný čas pracovníka: 10,000 Min.
Typ mzdy: Časová
Tarif: 542
Sadzba/hod: 2,32
Réžia/hod: 9,96
Mzda: 0,50
Réžia: 2,16
Cena op.: 2,66

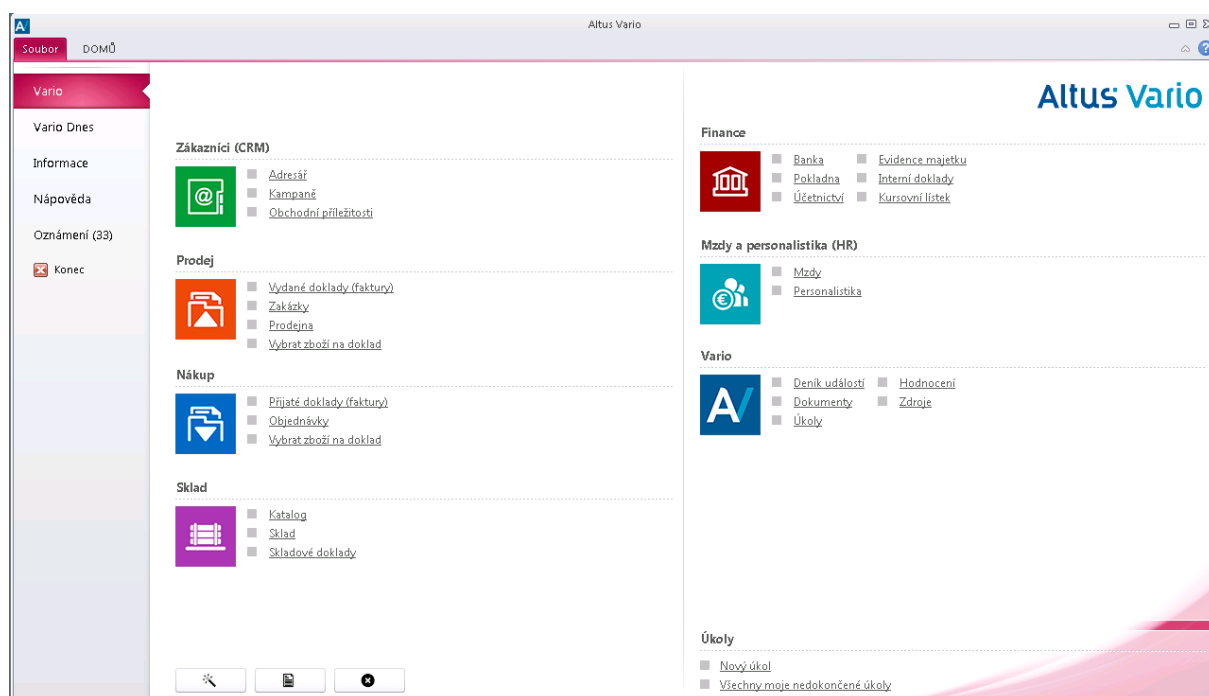
Popis operácie
K ustaveniu pripraviť poz.1-2ks, poz.2-3ks
V prípravku ustaviť dve tyce poz.1 , 3 ks.výstuhy poz.2
a pristehoval 6x zv.kút.2-10 E 5233,pr.2 AMP 80-100
Stehy ocistiť

OK Zrušenie Výkresy Upraviť výrobné záskazy Náradie Celková cena postupu: 43,92

Obr. 4-4 Addon Production [17]

4.2.4 Altus Vario

Altus Vario je produktem společnosti Altus software s.r.o.. Program je dostupný i v českém jazyce a je na trhu již od roku 2008. Filosofie uživatelské obsluhy softwaru využívá principy Microsoft Office Access, takže je snadno naučitelný. Program se nezabývá modifikacemi a vizualizacemi výroby a je orientován spíše ekonomicky než jako plánovač výroby. Tento software má cloudovou architekturu klient-server. Průměrná doba implementace u podniku střední velikosti je 1-3 měsíce. [19]



Obr. 4-5 Úvodní rozhraní Altus Vario [19]

4.2.5 Asprova

Produkt Asprova od stejnojmenné firmy je na dostupný na trhu od roku 2010. Tento software není příliš vhodný pro malé podniky, je orientován spíše na ty větší. V roce 2009 ovládala Asprova 48,8% japonského trhu softwaru s pokročilým plánováním a dodnes je považována v Japonsku na ztělesnění japonského know-how v oblasti plánování a řízení výroby.[20]

Asprova obsahuje algoritmy AATP,CTP i PTP, dá se v něm plánovat ve strategické, operativní i taktické úrovni. Asprova plánuje výrobu od čistě zakázkové až po velkosériovou. Případně je použitelný na plánování dle dopravy, dodávky nebo poptávky. Průměrná doba implementace u podniku střední velikosti jsou 3 měsíce.[21] Tento plánovací algoritmus pokrývá celé spektrum výroby, tedy od sériové až po zakázkovou výrobu.

Asprova je úzce provázán s MS Excelem, což usnadňuje kontrolu a zadávání dat. Také se tak snadno dá udělat plugin, stačí k tomu znalost maker v Excelu. Dále umí Asprova snadno sdílet data s dalšími programy jako MS Access, SQL servery, Oracle nebo textovými soubory. Software má certifikované rozhraní na největší ERP systémy – SAP, MS Dynamics, QAD, IBM.[22] Výhodou APS Asprova jsou propracované speciální funkce pro jednotlivé výrobní odvětví, tedy modulární systém. Tento software využívají například CANON Inc., Panasonic LCD nebo Toyota. [23]

Jelikož většina plánovačů z firem nejdříve pracuje s Excelem a teprve až s rostoucí různorodostí a objemem výrobků hledají profesionální alternativu pro plánování, určitě ocení podobnost, ne-li stejnost, funkcí Excelu s Asprovou. Další nesporná výhoda je možnost napojení Asprovy na stávající ERP systém.

4.2.6 Asprova jako plánovací software

Asprova nabízí několik modulů: MRP, BOM, MES, MS, APS, DS, NLS.

4.2.6.1 MRP a BOM

Moduly MRP (*Material Requirement Planning*) a BOM (*Bill of Material*) jsou vytvořeny pro výpočet potřeby materiálu bez možnosti optimalizace pořadí výroby.

4.2.6.2 MES

Modul MES (*Manufacturing Execution System*) obsahuje funkce k zobrazování aktuálního plánu a k získávání dat ze zpětné vazby z pracovišť. Tento modul je vytvořen hlavně pro pracoviště, takže zde lze nalézt veškeré výstupy v grafech a zprávách, ale nelze zde přepřepínávat. Druhá možnost použití je pro kontrolu aktuálního stavu výroby na pracovištích.

4.2.6.3 MS

MS (*Manufacturing Scheduler*) je určen pro plánování. A to pro dlouhodobé, střednědobé i krátkodobé. Plánování probíhá pro jednotlivá pracovní místa a sestavuje rovnou i pracovní pokyny. Vhodné používat pro jednotlivě plánované oblasti podniku.

4.2.6.4 APS

APS (*Advanced Planning and Scheduling*) modul je v podstatě sloučený MS modul s prodejními zakázkami a nákupem materiálu. APS je vhodný na jakékoli časové plánování. Modul zvládá plánování veškerých kroků od nákupu materiálu až pro expedici. Při plánování výroby vytváří detailní rozvrh pro veškeré výrobní procesy se započítáním omezení, jako například dostupnost materiálu nebo čekání na součást z jiného pracoviště.

4.2.6.5 DS

Jak už napovídá název modulu DS (*Data Server*), tento pomocný modul shromažďuje data ze všech modulů a formuje je.

4.2.6.6 NLS

NLS (*Network License Server*) je další pomocný modul, který umožňuje zpracovávání z každého počítače uvnitř datové sítě, aby bylo možné sdílení licencí v rámci sítě.[24]

4.2.6.7 Trial verze

K dispozici zdarma je vytvořena trial verze programu. Na základě této trial verze byl software dále hodnocen.

Po zvolení modulu APS se otevře uživatelské rozhraní působící trochu zastarale kvůli designu z Microsoft Office 95. To sice nemá vliv na funkčnost, ale nepůsobí to příliš důvěryhodně.

Firma Asprova má poměrně propracovaný systém pro uvedení zákazníka do fungování softwaru. K dispozici jsou instrukce na youtube.com, po objednání nebo zažádání o software jsou automaticky dodávány na zákazníkův email týdenní kurzy o práci s Asprovou.

4.2.7 Plánování v systému Asprova

4.2.7.1 Odvolávky

Po importování dat o výrobě (kusovníky, objednávky), je možné si otevřít přehlednou tabulku „Odvolávky“ o tom, co je třeba vyrobit. V tabulce lze vidět, co se bude vyrábět, pro koho, do kdy i v jakém množství. Po aktivování plánovacích funkcí Asprovy se k tabulce připojí další 2 údaje a to čas ukončení práce na objednávce a časová zásoba před expedicí produktu. Časová zásoba, takzvaný *buffer*, je opět barevně odlišena pro lepší přehled. Při výchozím nastavení

představuje zelená ideální buffer - 14 až 28 hodin, žlutá představuje mírný problém - rozmezí mezi 6 a 14 hodinami, kritický problém je znázorněn červenou barvou a nastává při bufferu menším než 6 hodin. Intervaly lze samozřejmě upravit.

Asprova odlišuje barevně buňky s informacemi o množství podle kvantity, podle předem nastaveného parametru.

4.2.7.2 Ganttův diagram

Jako další možné zobrazení je Ganttův diagram, velice oblíbená vizualizace ve většině APS a ERP systémech. Je to pruhový diagram znázorňující postup výroby v závislosti na čase. U Asprovy tedy představuje znázornění výroby produktů na jednotlivých pracovištích. Lze zde vyčíst od kdy, do kdy pracoviště dělá na jednom výrobku. Každý výrobek zde má svojí barvu pro rychlé odlišení, protože i když jsou zde popisky jednotlivých výrobků, tak při větší a rozmanitější výrobě jsou velice nepřehledné. Popisky jednotlivých výrobků obsahují kód výrobku, den, na který se vyrábí a zákazníka. Dále je zde vidět závislost výrobku na svých součástech, které se vyrábějí na jiném pracovišti, pauzy na pracovišti, nebo volná místa v době, kdy se na pracovišti nepracuje (noc, víkend).

Ganttův diagram se dá přepnout do přehlednější verze z hlediska času dodání, která barevně znázorňuje časové splnění zakázky. Červená značí nesplnění časové limitu, oranžová označuje splnění na hranici limitu, žlutá znamená mírné zpoždění, zelená je ideální stav a modrá signalizuje výrobu příliš dopředu, kde pak můžou vyvstat problémy se skladováním. Plánovač tedy může podle situace přidat, nebo naopak vypustit nějakou směnu na pracovišti. Asprova mu pak časové limity přepočítá.

4.2.7.3 Instrukce pro pracoviště

Jako výstup z plánovacího procesu, lze považovat tabulku úkolů pro jednotlivé pracoviště, kde je dáno kódové označení výrobků, přesný čas začátku a konce pracování na každém typu výrobku, kód projektu, množství daného typu výrobku a čas expedice.

Systém by měl být schopen vyšrafováním splněné úkoly na pracovištích. Lze tedy snadno vyčíst, kde pracoviště splnila plán, kde je pozadu nebo kde dokonce pracovní rozvrh předstihla.

5 Vstupy a výstupy

Na začátek by bylo vhodné vytvořit přehled základních funkcí každého softwaru, který se hlásí do skupiny APS systémů:

- Zohledňování dostupné kapacity výrobních zdrojů, tedy strojů, lidí a přípravků
- Možnost nadefinování určitých vzorků směnnosti pro výrobní zdroje
- Zohledňování dostupnosti materiálu i se zřetelem na pojistné zásoby, dodací lhůty a již potvrzené dodávky materiálu
- Funkce, která umožní vyškrtnout určitý zdroj z plánování (například kvůli plánovaným opravám strojů, dovolené)
- Funkce, která umožní fixní přiřazení výrobního zdroje v určitý čas pro určitou operaci (pro možnost překlenutí dvou plánovacích cyklů)
- Funkce pro definování úplných a částečných substitučních pracovišť v požadované účinnosti
- Funkce pro definování úplných nebo částečných substitučních materiálů pro speciální výrobky (případně díly)
- Funkce pro simulaci chování výrobního procesu pro nadefinované chování (více kusů výrobků než je aktuální stav, vnucení seřizování před každou výrobní dávkou nebo naopak seřizování jen u prvního kusu dávky, plánování operací, které musí jít bezprostředně za sebou apod.) [6]

5.1 Vstupy

Zavádění APS programů do podniku není jednoduchá záležitost, neboť se nejedná jen o napojení jakéhosi programu na databázi firmy, jak si často podniky představují a nutno dodat, že jim to leckterí obchodní zástupci nevyvracejí. Vstupy pro plánování a pro rozvrhování jsou skoro stejné. Rozdíl mezi nimi je takový, že poptávka a dodávka materiálu se řeší v plánování, kdežto rozvrhování pracuje s hotovými výrobními příkazy.

APS je velmi často implementováno do podniku s už zavedenými informačními systémy jako ERP, docházkový systém, různé CAD systémy. Nejtěžší část implementace APS proto bývá propojení stávajících informačních systémů s tím novým, protože starší systém nemusí být kompatibilní s novějšími systémy. Tyto propojení se provádí pomocí datových můstků. Systémy se musí sladit, aby bylo jasné, kdy bude probíhat výměna dat a který systém je bude přijímat a který odevzdávat, tedy kterým směrem data potečou. V neposlední řadě se proto musí ověřit, zda je podnik dostatečně hardwarově i softwarově vybaven, aby mohl zajistit data do následujících vstupů. [27]

5.1.1 Databáze

Samozřejmě, že kvalitní databáze je základ pro úspěšnou aplikaci APS do systému podniku. Databáze nemusí být vedena přímo v APS systému. Velmi často je totiž APS systém nastaven jen jako modul a pak čerpá informace z databáze příslušného ERP systému.

V těchto datech by měl být zaznamenán seznam objednávek od zákazníků pro informaci, co se má vyrobit a do kdy by se to mělo vyrobit. V praxi bývá vhodné dodat jistou preferenci zakázek, tedy jestliže se nemůžou stihnout vyrobit všechny zakázky včas, s kterou se můžeme pozdit, aby to firmě uškodilo co nejméně.

Dále by zde měl být vidět aktuální stav skladu a zároveň i objednávky materiálu s časem doručení, aby nedošlo k duplikaci objednávek na stejný materiál. Pokud jsou k výrobě potřeba i nějaké přípravy na doplňkové informace, měl by zde být přehled i o nich.

U rozpracovaných objednávek by měl být vidět jejich aktuální stav, tedy co už je hotovo, na čem se právě teď dělá a co bude třeba ještě udělat. Další neméně důležitá data, která by měla být v databázi zaznamenána, jsou pracoviště, které budou potřeba k zrealizování zakázek a dostupná pracoviště, která v podniku existují. V databázi by také měly být k dohledání všechny technologické postupy pro zjištění potřeby materiálu, pracovišť a času.

5.1.1.1 Databázová struktura

Drtivá většina APS systémů vyžaduje síťový databázový model, tedy model, kde se používají lineární i cyklické ukazatele, které vyjadřují vztahy mezi jednotlivými databázovými položkami. Tento model je velice složitý, kvůli velkému množství vazeb, takže se zde špatně implementují vnitřní změny databáze.[28]

5.1.2 Základní vstupy

5.1.2.1 Zdroje

Zdroje jsou zde charakterizovány jako omezení kapacity podniku. Jedná se o stroje, pracovní síly a nástroje. Také sem patří skladové prostory. Pro přehlednost bývají zdroje rozřazovány do různých skupin, kde se přiřazují podle společných vlastností, například přesnost stroje, zkušenost dělníka. Tento systém pak pomáhá rychleji a detailněji definovat co je potřeba k dané operaci.

5.1.2.2 Položka

Toto je nejčastěji používaný typ vstupu. Představuje nejen materiál, komponenty, polotovary i finální výrobky, ale také nástroje, náhradní díly pro stroje či různé přípravy. Z výčtu je vidět, že sem patří věci jak nakupované, tak vyrobené. Je ovšem pro další operace důležité tyto 2 skupiny odlišit dle informací o položkách, například při řešení velikosti pojistné zásoby.

5.1.2.3 Kusovník

Strukturovaný kusovník propojuje pracovní postupy a potřebu materiálů. Kusovník může být víceúrovňový i jednoúrovňový. To záleží na daném technologickém postupu výrobku.

5.1.2.4 Výrobní příkazy

Na výrobních příkazech je postavené rozvrhování. Výrobní příkaz je v podstatě již zaplánovaná zakázka, lze z něj vyčíst, které zdroje a materiál bude používat.

5.1.2.5 Požadavky

Požadavky mohou mít formu nadřazeného výrobního procesu, pojistné zásoby nebo případně jsou výsledkem prognóz.

5.1.2.6 Poptávky

Poptávky představují odběratelské zakázky nějaké položky. APS systémy jsou schopny počítat s objednacími dobami nakupovaných položek.

5.1.2.7 Dodávky

Dodávky představují distribuci vyrobených výrobků. Jsou charakterizovány nákupními objednávkami, množstvím na skladě a rozvrhem výroby.

5.1.2.8 Časové omezení

Jedná se o nastavení svátků a směn do systému (se zohledněním kapacit), pro správné zobrazení dostupnosti pracovní síly.

5.1.2.9 Parametry plánování

A konečně parametry plánování představují globální nastavení pro plánování v daném podniku. Jedná se o parametry jako plánování do omezených nebo neomezených kapacit nebo třeba horizont plánování, buffery a pojistné zásoby materiálu. [4]

5.1.2.10 Vedlejší vstupy

- Koeficient plnění norem
- Priorita kapacitní jednotky
- Časový interval pro přeplánování [29]

5.2 Výstupy

Většina APS modulů a systémů podává uživateli tyto základní výstupové informace:

- plánované ukončení výroby na zakázce
- plánované zpoždění u zakázek
- vygenerované příkazy s informací kdy a kde se budou provádět
- různé konflikty a nedostatky, jako nouze o materiál, úzká místa apod.
- případné simulace (viz níže)

Jelikož APS se překládá jako „Pokročilé plánování a rozvrhování“, je na místě poukázat na konkrétní výstupy plánování a konkrétní výstupy rozvrhování. Výstupem plánování je vygenerovaný příkaz s dvěma časovými informacemi: o začátku výroby a o jejím ukončení. Výstup rozvrhování je pracovní rozvrh, s počátečním a koncovým datem a se stanovenými prioritami jednotlivých úkonů.

5.2.1 Vedlejší výstupy

Ne každý uživatel APS se spokojí se základními výstupy. Ať už proto, že jeho výroba v jeho podniku je něčím specifická, nebo protože uvažuje stylem: „když už implementovat nový systém, tak ať stojí za to“. Proto je mnoho dalších možných vstupů, které sice necharakterizují APS systémy, ale mohou se k nim připojit a využívat jejich data k podávání po nich požadovaných informací.

- efektivita zařízení
- využití zdrojů
- kritická místa

- denní plán pro jednotlivá pracoviště
- kapacitní zatížení jednotlivých pracovišť
- automatické generování kapacitních kalendářů ve zvoleném časovém intervalu z kalendáře plánovacího a z druhu směrnosti kapacitní jednotky
- údržba kapacitních kalendářů
- časový interval pro přeplánování
- evidence zakázek pro každý den (i s plánovacími daty)

5.2.2 Simulace

Simulace je speciální výstup pro APS systémy, který se liší od „klasických“ výstupů tím, že pomáhá uživateli v rozhodování v určitých situacích. Uživatel si může nastavit své rozhodnutí, jako například rozhodnutí nakoupit polotovary, místo aby je sám vyráběl nebo přijmout zakázku navíc a zahájit přesčasové hodiny. Systém pak vyhodnotí tyto fiktivní rozhodnutí spolu s aktuální situací v podniku a poskytne výsledky, které vypovídají o tom, jak by se dané rozhodnutí projevilo. Je možné takovýchto fiktivních rozhodnutí, neboli scénářů, nasimulovat více a pak je mezi sebou porovnat, pro určení toho nejvýhodnějšího. Simulace zajišťuje funkce *what-if*. [4]

6 Algoritmy využívané v APS systémech

Jak již bylo zmíněno, předchůdcem APS je všeobecně brán MRP II. Následující algoritmy představují algoritmy APS, které, s jistou dávkou nadsázky, udělali onen pomyslný krok vpřed od MRP II a daly tím vzniknout APS systémům. Je třeba podotknout, že v praxi se pro APS používají kombinace těchto algoritmů.

6.1 ATP

ATP je zkratka od *Avialable to promise*, což přeloženo do češtiny znamená „závazná dodací lhůta“. Jedná se o nejstarší algoritmus pro APS. ATP je algoritmus používaný pro výrobu na sklad, tedy ve výrobě, kde se vyrábí dopředu a nečeká se na objednávku od zákazníka (ten si produkt vyzvedne rovnou z distribučního centra). Algoritmus zde vypočítává dodací lhůtu.

V případě, že je tento algoritmus používán u výroby na objednávku, určuje dobu výroby pomocí aktuálního stavu zásob, průběžných dob výroby a informace, co z produktu je již hotové. ATP je schopen počítat stav zásob v časové ose na všech úrovních skladu, tedy počítá s nákupem i odbytem. Výrobu navrhuje bez prodlev a předstihů. Algoritmus je charakteristický pro podniky s procesní výrobou.

6.2 ATTP

Algoritmus ATTP, tedy *Allocated avialable to promise*, je rozšířený ATP algoritmus. Má stejnou funkci jako ATP algoritmus a navíc je schopen rozdělovat hotové výrobky mezi jednotlivé zákazníky. Rozdělovat může podle vícera způsobů: buď podle priority zákazníka nebo zakázky, tedy jak důležitý je zákazník pro podnik a nebo podle geografické polohy. Algoritmus při rozdělování bere v potaz vynaložené náklady a možný zisk.

6.3 CTP

CTP je zkratkou pro *Capable to promise*, česky překládané jako „Kapacitní plánování výroby“. Tento algoritmus se využívá, pokud produkty z výroby či montáže nejsou k dispozici na skladě a vyrábí se přímo na zakázku. Algoritmus počítá s materiálem a výrobní kapacitou. Výstupem CTP je informace, kdy bude materiál a výrobní kapacita k dispozici po daný výrobek. CTP ovládá dopředné, zpětné i kombinované plánování. U tohoto algoritmu je možné vybrané zakázky zafixovat, což zajistí, že s nimi nebude v průběhu plánování přesouváno. Další schopnost algoritmu je volitelné nastavení chování při komplikacích, jako jsou například kolize materiálu.

6.4 PTP

PTP neboli *Profiteble to promise*, nejmladší ze základních algoritmů používaných v pokročilém plánování a rozhodování. Tento algoritmus v podstatě obsahuje prvky algoritmů ATP i algoritmů CTP. PTP sčítá náklady na výrobu a porovnává je s poptávkou a cenou za níž je zákazník ochoten si produkt koupit. Výstupem je tedy informace o tom, jaký přínos objednávka pro firmu představuje. Spojením algoritmů ATP a CTP tedy vznikl algoritmus vhodný pro všechny typy výroby. [28]

6.5 Frozen Intervals

Jedná se o genetické algoritmy pro dynamické pokročilé plánování a rozvrhování, pro které lze najít v literatuře pod zkratkou DAPS nebo d-APS.

Algoritmy se zabývají rozvrháním a vycházejí z toho, že dělat změny v rozvrhování na poslední chvíli může být nebezpečné, protože čím blíže se upravovaná událost blíží k aktuálnímu času, tím větší je šance, že změny zvýší náklady na výrobu a naopak sníží stabilitu plánu. Místo rozvrhování veškeré nezpracované práce při každé nové události budou přerozvrhovány jen ty úkoly, které jsou naplánované mimo určitý čas. Zbývající úkoly, tedy ty které by měli být aktuální během onoho určitého času, zůstanou nedotčené. Tyto nedotčené úkoly se nacházejí v takzvaném „zmraženém intervalu“, anglicky *frozen interval*. [30]

Autor této práce je nucen podotknout, že neví o žádném programu na české scéně APS systémů, který by tyto algoritmy využíval (proto je také zachován anglický název). S programem se setkal pouze v teoretické a simulační rovině.

7 Popis praktické části práce

Tato část práce je věnována popisu jednotlivých částí praktické části. Jak bývá i v jiných publikacích zvykem, písmem Courier je zaznamenán programový kód, programové výpisy, názvy datových typů a hodnoty vlastností. Všechny názvy proměnných jsou zachovány v originálních názvech, tedy tak, jak jsou použity v aplikaci.

7.1 Cíl praktické části práce

Dalším cílem této práce je vytvořit jednoduchou aplikaci, která by představovala určité nastínění některých funkcí APS systému. Aplikace neuvažuje omezení ze strany materiálu a ze strany kapacity strojů. Aplikace načte objednávkový list v podobě textového souboru a zpracuje ho. Výstupem je čistý čas jednotlivých pracovišť.

Aplikace byla vytvořena v prostředí Delphi. Při vývoji aplikace bylo bráno v potaz, že aplikace je vyvíjena pro imaginární firmu, zabývajícími se výrobou jízdnicích kol. Imaginární firma vyrábí svá kola na 5 pracovištích, která jsou pojmenována A,B,C,D,E. Počet pracovišť je konečný. Každé pracoviště je schopno v jeden čas pracovat pouze na jedné operaci.

7.2 Rozhraní formuláře

The screenshot shows a software window titled "Formulář". At the top, there is a label "Velikost měřítka:" followed by a text input field containing the number "1". To the right of this input are two buttons: "Načíst" and "Spustit". Below the input field, the interface is divided into two main sections. The first section is titled "Čistý čas pracovišť:" and contains five vertically stacked, colored buttons: blue ("Čistý čas Pracoviště A"), red ("Čistý čas Pracoviště B"), green ("Čistý čas Pracoviště C"), yellow ("Čistý čas Pracoviště D"), and magenta ("Čistý čas Pracoviště E"). The second section is titled "Celkový čas pracovišť:" and also contains five vertically stacked, colored buttons: blue ("Celkový čas Pracoviště A"), red ("Celkový čas Pracoviště B"), green ("Celkový čas Pracoviště C"), yellow ("Celkový čas Pracoviště D"), and magenta ("Celkový čas Pracoviště E"). In the bottom right corner of the window, there is a button labeled "Konec".

Obr. 7-1 Úvodní zobrazení formuláře

7.2.1 Popis formuláře z uživatelského pohledu

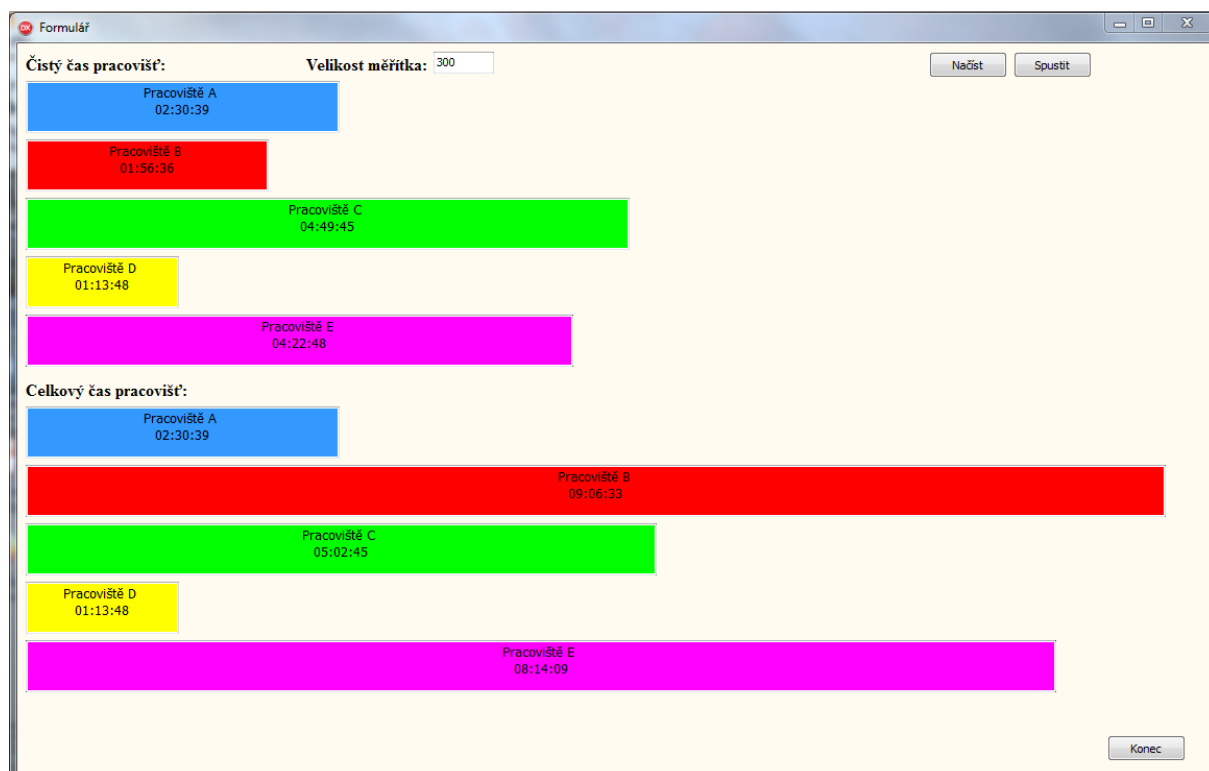
Na obrázku 7-1 lze vidět úvodní zobrazení tabulky. Většinu prostoru zabírá řada barevně odlišených polí, které mění svou velikost v závislosti na čase a na měřítku. Pole jsou rozdělena na dvě skupiny: *čistý čas pracovišť* a *celkový čas pracovišť*. V každé skupině jedno pole charakterizuje jedno z pracovišť. První skupina ukazuje, jak dlouho bude pracoviště

aktivně pracovat a druhá skupina ukazuje, kolik času uplyne, než se pracoviště dokončí zadanou práci. Časový rozdíl mezi poli znázorňující stejné pracoviště značí, kolik času bude pouze čekat na další materiál ke zpracování.

Dále formulář obsahuje 3 tlačítka a to jmenovitě *Načíst*, *Spustit* a *Konec*.

Uživatel by měl začít stisknutím tlačítka *Načíst*. Po stisknutí aplikace otevře nové okno pro vybrání souboru ve formátu XML představující objednávkový list. Po vybrání objednávkového listu je potřeba stisknout tlačítko *Spustit* pro výpočet časů na pracovištích. V případě, že objednávkový list nebyl načten nebo načtený objednávkový list nebude správně zformátovaný a načtený (nebude obsahovat klíčové položky), výpočet neproběhne, komponenty charakterizující pracoviště zmizí a zobrazí se chybová hláška (viz obrázek 7-3).

K ukončení aplikace slouží tlačítko *Konec*.



Obr. 7-2 Vizualizace jednoduchého plánu

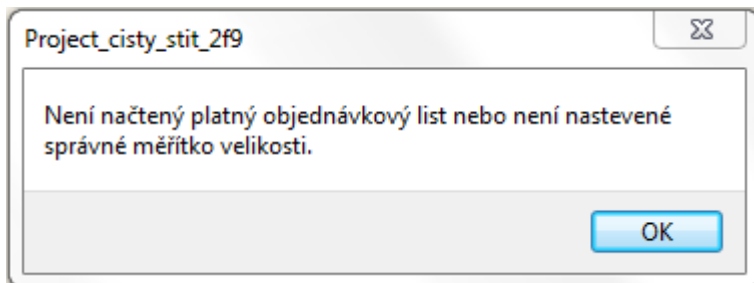
7.2.2 Popis formuláře z vývojářského pohledu

Jak lze vidět z obrázků 7-1 a 7-2, formulář obsahuje 10 komponent TMemo, 3 tlačítka TButton, 2 nápisy typu TLabel a 1 komponenta TEdit. Co na obrázku vidět nelze jsou 2 pomocné komponenty typu TMemo, do kterých se načítá soubor a jedna komponenta TOpenDialog, která ono načtení souboru umožňuje. Dále zde je ještě jeden informativní nápis typu TLabel, který je viditelný pouze při změně měřítka.

Viditelné komponenty typu TMemo představují pracoviště imaginární firmy, pro lepší vizuální přehled jsou jinak zabarveny. Počet pracovišť je pevně stanoven. V případě, že by bylo pracovišť více, nebo méně, musel by se programový kód upravit. Na každé pracoviště jsou přiřazeny 2 komponenty. Jedna komponenta z této dvojice představuje čistý čas, kdy se na pracovišti bude pracovat, a druhá komponenta představuje celkový časový nárok na

pracoviště. Tedy celkový čas dohromady s nutnými časovými mezerami kvůli technologičnosti výroby.

Při opakovaném používání stejného souboru sloužícího jako objednávkový list je připojena k programu procedura, která načte soubor s pevně danou cestou. Nicméně aktivovat se musí přímo v kódu.



Obr. 7-3 Chybné načtení objednávkového listu

7.3 Objednávkový list

Objednávkový list zde představuje soubor, který obsahuje informace o objednávkách a následně o výrobních údajích jednotlivých komponent objednaného produktu. Jedná se o soubor ve značkovacím jazyce XML. Příklad objednávkového listu je možno nalézt v příloze A.

Každá objednávka obsahuje tyto údaje:

- Zákazník
- Jméno objednaného výrobku
- Kód výrobku
- Cena výrobku
- Počet objednaných kusů
- Kusovník

Pro aplikaci jsou údaje zákazník, jméno výrobku, kód výrobku a cena výrobku informativní položky.

Položka Kusovník se skládá ze seznamu dílů potřebných pro sestavení objednaného produktu. Jednotlivé díly v sobě nesou údaje o kódu dílu a výrobním postupu dílu. Kód dílu je pro aplikaci další informativní položka a dále není zpracováván. Výrobní postup obsahuje informace o tom, které pracoviště jsou k vyrobění potřeba a jaký je čas potřebný na práce na každém pracovišti. Čas dopravy polotovaru nebo materiálu k pracovišti je zahrnut do času práce. Pro lepší představení a pochopení objednávkového listu je vložen příklad části jedné objednávky. Veškeré údaje od zákazníka po jednotlivé časy jsou smyšlené.

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<Objednavkovy_list>
  <Objednavka Zakaznik="Strakal">
    <name>Horske kolo Freiman</name>
    <cena>15.375Kc</cena>
    <symbol>HK_Frm</symbol>
    <pocet>3</pocet>
    <Sestava>
      <Podsestava1>Predni vidlice
        <kolo>
          <znak>K55G</znak>
          <vyroba>
            <pracoviste A>
              <cas>00:02:27</cas>
            </pracoviste A>
            <pracoviste B>
              <cas>00:01:12</cas>
            </pracoviste B>
          </vyroba>
        </kolo>
        <vidlice>
          <znak>vid5536A1</znak>
          <vyroba>
            <pracoviste A>
              <cas>00:01:53</cas>
            </pracoviste A>
            <pracoviste C>
              <cas>00:21:18</cas>
            </pracoviste C>
          </vyroba>
        </vidlice>
      <Ram>
        <znak>RamVid5520A1</znak>
        <vyroba>
          <pracoviste D>
            <cas>00:12:18</cas>
          </pracoviste D>
          <pracoviste C>
            <cas>00:12:36</cas>
          </pracoviste C>
          <pracoviste B>
            <cas>00:07:36</cas>
          </pracoviste B>
        </vyroba>
      </Ram>
    </Sestava>
  </Objednavka>
</Objednavkovy_list>
```

Obr. 7-4 Ukázka z objednáčního listu

7.4 Vnitřní stavba aplikace

7.4.1 Parsování

Dopředné plánování zde funguje tak, že priorita objednávek je určena pozicí v kusovníku. Stejně tak se děje s jednotlivými výrobky a i s jednotlivými díly. Například tedy z předchozího objednáčního listu (viz obrázek 6-4) je zřejmé, že díl *K55G* má vyšší prioritu než díl *Vid5536A1* a díl *RamVid5520A1*. Technologický postup výroby dílů zde představuje seznam po sobě jdoucích pracovišť. Aplikace počítá s pevným nebo nulovým časem seřizování stroje a dopravy materiálu z jednoho pracoviště na druhé.

V praxi to znamená, že výroba vycházející z části objednávkového listu uvedeného jako příklad začne naplánováním veškerých dílů *K55G* před ostatními díly. Po naplánování všech dílů *K55G* se začne plánovat s díly *Vid5536A1*.

Při naplánování dílů *K55G* vzniknou zákonitě na všech pracovištích kromě prvního časové mezery, protože pracoviště bude čekat, až se k nim obráběný materiál dostane. Při plánování

každého dalšího dílu se aplikace pokusí vzniklé mezery zaplnit, pokud to dovoluje časová náročnost dané operace a zároveň pokud to dovoluje technologický postup.

Algoritmus se sestává z určitého začátku, tedy načtení správných dat a pak z prakticky 5 totožných bloků. Každý tento blok se věnuje jednomu pracovišti.

Po zjištění, že uvedený text je opravdu čas, následuje zapsání času samotného. Jak lze vypočítat z ukázky objednacího listu, je zapsání času pevně dané. Čas se zapisuje hned za lomenou závorku a to ve tvaru *hh:mm:ss*. Pomocí těchto pravidel pak není problém čas vyjmout a načíst do správných proměnných. Pro uvedenou ukázkou platí že *secAA* představuje čas potřebný na danou operaci. Tento čas je uváděný v sekundách. Proměnná *ks* určuje, kolik kusů daného dílu se bude vyrábět.

```
secAA:=(10*3600*StrToInt(pole2[m+5])+3600*StrToInt(pole2[m+6])+10*60  
*StrToInt(pole2[m+8])+60*StrToInt(pole2[m+9])+10*StrToInt(pole2[m+11  
) +StrToInt(pole2[m+12]))*ks;
```

Po načtení času dojde k samotnému dopřednému plánování. Pro lepší orientaci čtenáře budou ale nejdříve představeny proměnné, které v plánování figurují.

7.4.2 Využití proměnné

Nejdříve budou představeny proměnné, které jsou identické pro každé pracoviště. O tom, ke kterému pracovišti proměnné patří, vypovídají velká písmena v názvu proměnné. V příkladech budou zobrazovány jen části algoritmu pro *pracoviště A*. Jedná se o proměnné *secAA*, *casA* a pole *PcasA*.

SecAA již bylo představeno. Jedná se o čas dané operace, tedy proměnná je pro každou operaci jiná.

Proměnná *casA* je celkový čas pracoviště. Tato proměnná tedy ukazuje, jak dlouho bude muset být pracoviště spuštěné, aby se splnily všechny objednávky v objednacím listě.

PcasA je pole, kam se zapisují časové mezery vzniklé při čekání pracoviště na materiál.

Druhá skupina proměnných platí pro všechny pracoviště dohromady. Jedná se o proměnné *cov*, *CPO*, *kontrola*, *První*, *Poslední*, *k*, *kA*.

Proměnná *cov* je zkratka od „číslo operace výrobku“ a představuje počítadlo operací pro daný díl.

CPO je zkratka od „čas poslední operace“ a skutečně zaznamenává čas operace nacházející se bezprostředně před právě plánovanou operací. *CPO* tedy představuje časovou hranici, od které aplikace může začít danou operaci plánovat.

Proměnná *kontrola* ukazuje, zda už došlo k uložení času.

Do proměnných *První* a *Poslední* se při speciálním případě ukládají začátek a konec časové mezery (viz níže).

Ve stejném speciálním případě se také objevuje proměnná *k*, která slouží čistě jako pomocná proměnná *for cyklu*.

Proměnná *kA* je také pomocná proměnná. Určuje pozici v poli pro časové mezery. Proměnná *kA* je stejná pro všechny pracoviště, i když je označena velkým písmenem „A“. Autor tak ctí zásadu „podmínka potvrzuje pravidlo“.

7.4.3 Dopředné plánování

Ted' již k samotnému algoritmu dopředného plánování. Algoritmus každého pracoviště se skládá ze dvou částí. Ta menší část se zabývá případem, kdy se jedná o první operaci dílu. Aplikace se jednoduše pokusí operaci naplánovat do nejbližší možné časové mezery, pokud žádná časová mezera není dost velká, aby se tam daná operace vešla, zařadí aplikace danou operaci na konec výrobní fronty.

```
if cov=1 then
begin
kA:=1;
kontrola:=0;
repeat
if secAA < (PcasA[kA+1]-PcasA[kA]) then
begin
CPO:=PcasA[kA]+secAA;
PcasA[kA]:=PcasA[kA]+secAA;
kA:=(length(PcasA)-1);
inc(kontrola);
end
else
kA:=kA+2;
until kA=(length(PcasA)-1);
if kontrola=0 then
begin
casA:=casA+secAA;
CPO:=casA;
end;
end
```

Druhá část pak řeší situaci, kdy daná operace není první operací a je tady třeba počítat s proměnnou CPO. Všeobecně můžou nastat 3 situace.

Čas skončení prací na předchozí operaci dílu (CPO) je menší než čas skončení poslední operace na daném pracovišti (casA). Čas právě plánované operace je dostatečně malý na to, aby se vešel do některé z časových mezer pracoviště, které se nachází na časové ose za časem CPO. V této situaci se operace naplňuje do mezery. Daná mezera se zmenší o příslušný čas. Případně, pokud je kvůli CPO operace naplánována doprostřed mezery, se časová mezera rozdělí na 2 menší. K tomuto speciálnímu případu se používají proměnné První, Poslední a k.

Druhá možnost se liší od první pouze tím, že se operace do žádné časové mezery nevejde a tak se naplňuje na konec výrobní fronty.

Třetí situace nastává, když CPO je větší než casA. V tomto případě aplikace naplňuje právě plánovanou operaci od okamžiku, kdy skončí práce na předchozí operaci dílu (s případným přičtením seřizování a dopravy materiálu na pracoviště). Tím vznikne nová časová mezera, která může být později zaplněna nějakou další operací.

```
else
begin
if casA > CPO then
begin
```

```
kA:=1;
kontrola:=0;
repeat
if CPO < PcasA[kA+1] then
begin
if CPO > PcasA[kA] then
begin
if secAA < (PcasA[kA+1]-CPO) then
begin
Prvni:=PcasA[kA];
Posledni:=CPO;
PcasA[kA]:=CPO+secAA;
CPO:=CPO+secAA;

for k:=(length(PcasA)-2) downto kA do
begin
PcasA[k+2]:=PcasA[k];
end;

PcasA[kA]:=Prvni;
PcasA[kA+1]:=Posledni;
kA:=(length(PcasA)-1);
inc(kontrola);
end;
end;

if CPO = PcasA[kA] then
begin
if secAA < (PcasA[kA+1]-CPO) then
begin
PcasA[kA]:=CPO+secAA;
CPO:=CPO+secAA;
kA:=(length(PcasA)-1);
inc(kontrola);
end;
end;
if CPO < PcasA[kA] then
begin
if secAA < (PcasA[kA+1]-PcasA[kA]) then
begin
CPO:=PcasA[kA]+secAA;
PcasA[kA]:=PcasA[kA]+secAA;
kA:=(length(PcasA)-1);
inc(kontrola);
end
end;
end;
if kontrola=0 then kA:=kA+2;
until kA=(length(PcasA)-1);
if kontrola=0 then
begin
casA:=casA+secAA;
CPO:=casA;
end;
end;
```

```
    if casA < CPO then
        begin
            kA:=0;
            repeat
                inc (kA);
            until PcasA[kA]=-1;
            PcasA[kA]:=casA;
            PcasA[kA+1]:=CPO;
            casA:=CPO+secAA;
            CPO:=casA;
        end;
    if casA = CPO then
        begin
            CPO:=casA+secAA;
            casA:=casA+secAA;
        end;
    end;
end;
```

7.5 Nedostatky a zjednodušení dopředného plánování

Jelikož má práce sloužit jako nastínění některých funkcí APS systémů, byly některé aspekty zjednodušeny nebo přímo vynechány. Dělo se tak buď z důvodu složitosti implementace daného problému, nebo naopak z důvodů přílišné jednoduchosti. Tedy, že se jedná o věc tak samozřejmou, že není třeba jí ukazovat.

V případě, že pro výrobek je potřeba některý díl ve větším množství než jeden kus, musí se daný díl zadat do objednávkového listu tolikrát, kolikrát je potřeba. Jednotlivé díly lze samozřejmě zadat s libovolnou prioritou. Aplikace pak podobnost dílů vůbec neřeší a počítá s nimi jako se všemi jinými díly. Tento problém však lze vyřešit snadným přidáním atributu „kusy“ u každého dílu. Atribut „kus“ již je implementována u každé objednávky, takže by se tento postup opakoval.

Vytváření objednávkového listu v textovém editoru není pohodlné a ani hodné jakékoli seriózní aplikace. Tento problém lze vyřešit sestavením jednoduchého formuláře, tak aby se jednotlivá data zapisovala. Těžší práce by již byla se sestavením určité knihovny dílů, která by obsahovala všechny potřebné informace. Ve formuláři by se pak zadávaly pouze požadované výrobky a počet kusů (případně zákazník).

Aplikace také postrádá funkci, která by v případě časového konfliktu několika dílů na jednom pracovišti upřednostnila výrobu dílu, který patří k zakázce mající vyšší prioritu než ostatní zakázky. Tato funkce by šla vytvořit jednoduchou porovnávací rovnicí a rozšířením objednávkového listu o atribut „priorita zakázky“.

Mezi další zjednodušení patří zanedbání omezení ze strany skladu a celkové zjednodušení problémů s distribucí materiálů na jednotlivá pracoviště.

Aplikace uvažuje nulový čas nebo pevný čas dopravy na pracoviště a výrobní dávka pro pracoviště je vždy rovna počtu objednaných kusů. Aplikace tedy neumožňuje paralelní operace na jednom dílu. Tento problém jde vyřešit detailnějším kusovníkem jednotlivých dílů a layoutem pracovišť.

Rovněž byla zanedbána zmetkovitost, program počítá s ideální výrobou beze zmetků.

8 Závěr

Kvalitní informační systém je dnes nepostradatelnou součástí každé moderní organizace a význam informačních technologií v současné době exponenciálně narůstá. Kromě komplexních řešení se stávají oblíbené právě aplikace na řízení výroby. Mnoho firem a společností má výrobní plány velmi rozmanité a plánování výroby bez kvalitní softwarové podpory se pro ně stává noční můrou. Přestože mají tyto aplikace i své odpůrce, stoupající poptávka na trhu hovoří za vše.

Projekt byl pojat jako studie pokročilého plánování. Cílem bylo, aby čtenář měl po přečtení hrubý přehled o obecné funkčnosti a o tuzemském trhu s APS systémy. V první části je čtenář seznámen s problematikou a vymezením ASP, důvodem jeho vzniku a zavedení do podniků a s vzorkem českého trhu a s podrobnější analýzou systému Asprova i s postupem pro jeho využití.

Další část práce se věnuje vnitřnímu fungování APS systémů a to konkrétně obecným vstupům a výstupům APS softwarům a jejich základním algoritmům.

Součástí této práce je aplikace, která představuje nastínění některých funkcí APS programů. Část práce se tedy věnuje popisu aplikace a podrobnému vysvětlení fungování algoritmu pro dopředné plánování.

Jako možné pokračování této práce se nabízí rozpracování zmíněné aplikace v plnohodnotný APS systém.

9 Seznam použitých zdrojů

- [1] DOSTÁL, Petr. *Pokročilé metody analýz a modelování v podnikatelství a veřejné správě*. 2008. ISBN 978-80-7204-605-8.
- [2] APS. *Svetproduktivity* [online]. [cit. 2016-10-29]. Dostupné z: <http://www.svetproduktivity.cz/slovník/APS.htm>
- [3] Production Planning System (PPS) a Advanced Planning and Scheduling (APS). *CIE* [online]. [cit. 2016-11-11]. Dostupné z: <http://www.cie-plzen.cz/index.php/cz/lexikon-metod/production-planning-system-pps-a-advanced-planning-and-scheduling-aps>
- [4] FERENČÍKOVÁ, Denisa. *Projekt optimalizace systému plánování výroby v ROSTRA s.r.o. pro efektivní zavedení APS aplikace*. Univerzita Tomáše Bati v Zlíně, 2009.
- [5] Pokročilé plánování a rozvrhování APS. *Iteuro* [online]. 2014 [cit. 2016-11-26]. Dostupné z: <http://www.iteuro.cz/produkty/aps-pokrocile-planovani/>
- [6] VELKOBORSKÝ, Jan. *Analýza úspěšnosti implementace APS systému za podpory V-A-T analýzy*. Západočeská univerzita, 2001.
- [7] PRÁŠIL, Zdeněk. *Analýza trhu s aplikacemi a technologiemi APS/SCM*. Vysoká škola ekonomická v Praze, 2009.
- [8] ŠTŮSEK, Jaromír. *Řízení provozu v logistických řetězcích*. 2007. ISBN 978-80-7179-534-6.
- [9] *Anketa: Zavádění APS systémů v tuzemském průmyslu* [online]. 2010 [cit. 2016-11-20]. Dostupné z: <https://www.systemonline.cz/rizeni-vyroby/anketa-zavadeni-aps-systemu-v-tuzemskem-prumyslu-1.htm>
- [10] IVERT, LINEA KJELSDOTTER. *Use of Advanced Planning and Scheduling (APS) systems to support manufacturing planning and control processes*. Göteborg, Sweden, 2012. CHALMERS UNIVERSITY OF TECHNOLOGY.
- [11] APS systém nenahradí funkcionalitu ERP systému. *Systemonline* [online]. 2010 [cit. 2016-12-03]. Dostupné z: <https://www.systemonline.cz/rizeni-vyroby/aps-system-nenahradi-funkcionalitu-erp-systemu.htm>
- [12] Simio. *Capterra* [online]. 2015 [cit. 2016-10-01]. Dostupné z: <http://www.capterra.com/production-scheduling-software/spotlight/145217/Simio/Simio>
- [13] Simio Production Scheduling Software. *Simio* [online]. [cit. 2016-10-01]. Dostupné z: <http://www.simio.com/products/production-scheduling-software.php>
- [14] *Plánování a řízení výroby - APS* [online]. [cit. 2016-10-08]. Dostupné z: <http://www.karatsoftware.cz/erp-karat/funkcionalita/planovani-rizeni-vyroby-aps/?gclid=CLPIpsD3u88CFRYo0wod8NEFKw>
- [15] *Karat* [online]. [cit. 2016-10-08]. Dostupné z: <https://www.systemonline.cz/prehled-informacnich-systemu/aps-systemy/karat-4.htm>
- [16] APS Karat Gantt. In: *Karatsoftware* [online]. [cit. 2016-12-03]. Dostupné z: <http://www.karatsoftware.cz/Files/FckGallery/16-karat-aps-gantt.jpg>
- [17] *AddOn Production* [online]. [cit. 2016-10-08]. Dostupné z: <http://www.gtsystems2.sk/sk/Article/293/AddOn%20Production.proxia>

- [18] AddOn Production. *Systemonline* [online]. 2014 [cit. 2016-10-08]. Dostupné z: <https://www.systemonline.cz/prehled-informacnich-systemu/aps-systemy/addon-production-2.htm>
- [19] Altus Vario. *Systemonline* [online]. 2014 [cit. 2016-10-08]. Dostupné z: <https://www.systemonline.cz/prehled-informacnich-systemu/aps-systemy/altus-vario-1.htm>
- [20] Analogie řízení výroby s pohybem lidského těla - využití APS a MES systému. *Svetproduktivity* [online]. 2011 [cit. 2016-10-29]. Dostupné z: <http://www.svetproduktivity.cz/clanek/Analogie-rizeni-vyroby-s-pohybem-lidskeho-tela--vyuziti-APS-a-MES-systemu.htm>
- [21] Asprova. *Systemonline* [online]. 2014 [cit. 2016-10-08]. Dostupné z: <https://www.systemonline.cz/prehled-informacnich-systemu/aps-systemy/asprova.htm>
- [22] *AIM_Asprova_CZ* [online]. 2016 [cit. 2016-10-08]. Dostupné z: http://www.aimtec.cz/wp-content/uploads/sites/133/2016/08/AIM_Asprova_CZ.pdf
- [23] Flexibility Through Excellent Integration Capabilities and Plug-in System. *Asprova* [online]. [cit. 2016-10-08]. Dostupné z: <http://www.asprova.eu/en/get-to-know-asprova/system-integration/>
- [24] Systémové moduly. *Asprova* [online]. 2015 [cit. 2016-10-29]. Dostupné z: <http://www.asprova.cz/moduly-a-funkce/systemove-moduly>
- [25] APS systémy. *Business world*. 2000, **2000**(8), 6.
- [26] Production Planning System (PPS) a Advanced Planning and Scheduling (APS). *CIE* [online]. [cit. 2016-11-11]. Dostupné z: <http://www.cie-plzen.cz/index.php/cz/lexikon-metod/production-planning-system-pps-a-advanced-planning-and-scheduling-aps>
- [27] PLÁČEK, Petr. Předprojektová analýza při implementaci APS systémů. *Systemonline.cz* [online]. [cit. 2017-03-14]. Dostupné z: <https://www.systemonline.cz/rizeni-vyroby/predprojektova-analyza-pri-implementaci-aps-systemu.htm>
- [28] ZÁPOTOČNÝ, Václav. *Aplikace APS systému pro plánování a rozvrhování výroby*. Brno, 2014. Brno University of Technology.
- [29] EVIS/400-APS modul. *Inco, spol.* [online]., 4 [cit. 2017-03-14]. Dostupné z: https://www.inco.cz/files/docs/inco_aps.pdf
- [30] MASUCHUN, Ruedee a Wiboon MASUCHUN. *New Rescheduling Methodology with Frozen Interval*. Bangkok. King Mongkut's Institute of Technology Ladkrabang.
- [31] KOPEČEK, P., PINTE, M. Optimization Heuristics for Supplies of Assembly Lines. *Periodica Polytechnica Transportation Engineering*, 2014, roč. 42, č. 1, s. 49-52. ISSN: 0303-7800

10 Seznam Příloh

Příloha 1.....Programový kód aplikace

Příloha 2..... Objednávkový list

Příloha 1

Programový kód aplikace

Katedra průmyslového inženýrství a managementu

Jan Průcha

```
unit Unit_Cisty_Stit_2f9;

interface

uses
  Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants,
  System.Classes, Vcl.Graphics,
  Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls;

type
  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Memo1: TMemo;
    MemoA: TMemo;
    MemoB: TMemo;
    MemoC: TMemo;
    MemoD: TMemo;
    MemoE: TMemo;
    OpenDialog1: TOpenDialog;
    Button3: TButton;
    Memicko: TMemo;
    MemoAA: TMemo;
    MemoBB: TMemo;
    MemoCC: TMemo;
    MemoDD: TMemo;
    MemoEE: TMemo;
    EditKonst: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    LabelMeritko: TLabel;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure EditKonstChange(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation
  procedure hodinky(hod,min,sec:integer;memo:Tmemo);
  var polehod,polemin,polesec:string;
  begin
    if hod < 10 then polehod:='0'+IntToStr(hod)
    else polehod:=IntToStr(hod);
    if min < 10 then polemin:='0'+IntToStr(min)
    else polemin:=IntToStr(min);
    if sec < 10 then polesec:='0'+IntToStr(sec)
    else polesec:=IntToStr(sec);

    memo.Text:=polehod+':'+polemin+':'+polesec;
  end;
```

Katedra průmyslového inženýrství a managementu

Jan Průcha

```
procedure memicka(pole:array of LongInt; memmo:TMemo); //pro zápis
časových mezer - nepoužívá se
var i:integer;
    pole5:string;
begin
    pole5:='';
    for i:=1 to 400 do
    begin
        pole5:=pole5+InttoStr(pole[i-1])+ ' a ';
    end;

    memmo.text:=(pole5);
end;

procedure vyskamemicka(cas,k:integer; memmo:TMemo); //používá se
begin
    memmo.Height:=(10*cas) div k;
end;

    procedure sirkamemicka(cas,k:integer; memmo:TMemo); //používá se
begin
    memmo.width:=(10*cas) div k;
end;

function casvsec(pole:string;m,kusy:integer):LongInt; //používá se (u
prac. A ne)
begin
result:=(10*3600*StrToInt (pole [m+5])+3600*StrToInt (pole [m+6])+10*60*StrToIn
t (pole [m+8])+60*StrToInt (pole [m+9])+10*StrToInt (pole [m+11])+StrToInt (pole [m
+12]))*kusy;
end;

function Nacist(const FileName: TFileName): String;
//používá se pro načtení pevně daného souboru
var
    FileStream : TFileStream;
    Bytes: TBytes;

begin
    Result:= '';
    FileStream:= TFileStream.Create(FileName, fmOpenRead or
fmShareDenyWrite);
    try
        if FileStream.Size>0 then begin
            SetLength(Bytes, FileStream.Size);
            FileStream.Read(Bytes[0], FileStream.Size);
        end;
        Result:= TEncoding.ASCII.GetString(Bytes);
    finally
        FileStream.Free;
    end;
end;

procedure TForm1.Button3Click(Sender: TObject); //používá se pro
načtení libovolného souboru
var policko:string;
begin

    if OpenFileDialog1.Execute then policko:=Nacist(OpenDialog1.FileName);
```

Katedra průmyslového inženýrství a managementu

Jan Průcha

```
    memol.text:=policko;

end;

procedure TForm1.EditKonstChange(Sender: TObject);
begin
LabelMeritko.Visible:=true;
end;

procedure TForm1.Button1Click(Sender: TObject);
//alfa omega aplikace
const min=1;
      max=400;
var
    konst:integer;

i, j, k, l, m, n, o, x, w, ww, A, B, C, D, E, hodA, minA, secA, hodB, minB, secB, hodC, minC, secC
, hodD, minD, secD, hodE, minE, secE: integer;
    pocetobjednavek, ks, kontrola, kA, Prvni, Posledni: integer;
//dopředné plánování+kusy
    secAA, casA, secBB, casB, secCC, casC, secDD, casD, secEE, casneE: LongInt;
//dopředné plánování

Asec, Amin, Ahod, Bsec, Bmin, Bhod, Csec, Cmin, Chod, Dsec, Dmin, Dhod, Esec, Emin, Ehod:
integer;
    CPO: integer; //čas poslední operace
    cov: integer; //číslo operace výrobku
    pole: string;
    PcasA, PcasB, PcasC, PcasD, PcasE: array[min..max] of LongInt;
//časové mezery pracovišť
    pole2, pole3, pole4, pole5: string;

begin
    //pole2:=Nacist('C:\Users\Prucha\Documents\H\FST\Advanced Planning and
Scheduling\Praktická část\Kusovník.txt');
    //pole2:=Nacist('C:\Users\Prucha\Documents\H\FST\Advanced Planning and
Scheduling\Praktická část\Kusovník - kopie.txt');
    pole2:=memol.text;
//nastavení začátečních hodnot
    casA:=0;
    casB:=0;
    casC:=0;
    casD:=0;
    casneE:=0;
    for k:=1 to max do begin PcasA[k]:=-1; end;
    PcasB:=PcasA;
    PcasC:=PcasA;
    PcasD:=PcasA;
    PcasE:=PcasA;

    hodA:=0;
    minA:=0;
    secA:=0;
    hodB:=0;
    minB:=0;
    secB:=0;
    hodC:=0;
    minC:=0;
    secC:=0;
    hodD:=0;
```

Katedra průmyslového inženýrství a managementu

Jan Průcha

```
minD:=0;
secD:=0;
  hodE:=0;
minE:=0;
secE:=0;
n:=0;

pole3:='';
pole4:='';
pocetobjednavek:=0;

for i:=1 to (length(pole2)-19) do //objednávka
begin
  if ord(pole2[i])=ord('<') then
  if ord(pole2[i+1])=ord('O') then
  if ord(pole2[i+2])=ord('b') then
  if ord(pole2[i+3])=ord('j') then
  if ord(pole2[i+4])=ord('e') then
  if ord(pole2[i+5])=ord('d') then
  if ord(pole2[i+6])=ord('n') then
  if ord(pole2[i+7])=ord('a') then
  if ord(pole2[i+8])=ord('v') then
  if ord(pole2[i+9])=ord('k') then
  if ord(pole2[i+10])=ord('a') then
  if ord(pole2[i+11])=ord(' ') then
  begin
    ks:=0;

    inc(pocetobjednavek);

    for l:=i to (i+200) do //pocet kusu
      if ord(pole2[l])=ord('<') then
      if ord(pole2[l+1])=ord('p') then
      if ord(pole2[l+2])=ord('o') then
      if ord(pole2[l+3])=ord('c') then
      if ord(pole2[l+4])=ord('e') then
      if ord(pole2[l+5])=ord('t') then
      if ord(pole2[l+6])=ord('>') then
      begin
        n:=0;
        pole3:='';
        repeat
          begin
            inc(n);
          end;
        until ord(pole2[l+7+n])=ord('<');
        for x:=1 to n do
          begin
            pole3:=pole3+pole2[l+6+x];
            ks:=StrToInt(pole3)
          end;
        end;
        pole4:=pole4+'objednavka č.'+IntToStr(pocetobjednavek)+' - pocet
kusu: '+IntToStr(ks)+' _____';
        k:=0;
        w:=0;
        j:=i+11;
        while w=0 do
          begin
            if ord(pole2[j])=ord('<') then
```

```
if ord(pole2[j+1])=ord('/') then
if ord(pole2[j+2])=ord('O') then
if ord(pole2[j+3])=ord('b') then
if ord(pole2[j+4])=ord('j') then
if ord(pole2[j+5])=ord('e') then
if ord(pole2[j+6])=ord('d') then
if ord(pole2[j+7])=ord('n') then
if ord(pole2[j+8])=ord('a') then
if ord(pole2[j+9])=ord('v') then
if ord(pole2[j+10])=ord('k') then
if ord(pole2[j+11])=ord('a') then
if ord(pole2[j+12])=ord('>') then
begin
  w:=1;
  inc(A);
end;

if (j+21)>length(pole2) then w:=1;

if ord(pole2[j])=ord('<') then //výroba
if ord(pole2[j+1])=ord('v') then
if ord(pole2[j+2])=ord('y') then
if ord(pole2[j+3])=ord('r') then
if ord(pole2[j+4])=ord('o') then
if ord(pole2[j+5])=ord('b') then
if ord(pole2[j+6])=ord('a') then
if ord(pole2[j+7])=ord('>') then
begin
  ww:=0;
  o:=j;
  CPO:=0;
  cov:=0;
  while ww=0 do
  begin
    if ord(pole2[o])=ord('<') then
      if ord(pole2[o+1])=ord('/') then
        if ord(pole2[o+2])=ord('v') then
          if ord(pole2[o+3])=ord('y') then
            if ord(pole2[o+4])=ord('r') then
              if ord(pole2[o+5])=ord('o') then
                if ord(pole2[o+6])=ord('b') then
                  if ord(pole2[o+7])=ord('a') then
                    if ord(pole2[o+8])=ord('>') then
                      begin
                        ww:=1;
                      end;
                    if (o+21)>length(pole2) then ww:=1;
                  if ord(pole2[o])=ord('<') then
//vyber pracoviste
if ord(pole2[o+1])=ord('p') then
if ord(pole2[o+2])=ord('r') then
if ord(pole2[o+3])=ord('a') then
if ord(pole2[o+4])=ord('c') then
if ord(pole2[o+5])=ord('o') then
if ord(pole2[o+6])=ord('v') then
if ord(pole2[o+7])=ord('i') then
if ord(pole2[o+8])=ord('s') then
if ord(pole2[o+9])=ord('t') then
if ord(pole2[o+10])=ord('e') then
```

Katedra průmyslového inženýrství a managementu

Jan Průcha

```
if ord(pole2[o+11])=ord(' ') then
    begin
        inc(cov);
        //if cov<>1 then CPO:=CPO+120;    //čas manipulace materiálu, 120
lze nahradit konstatnou
        //A
        if ord(pole2[o+12])=ord('A') then
            begin
                //CPO:=CPO+180    //čas seřizení stroje, 180 lze nahradit
konstatnou
                for m:=o to (o+20) do

                    begin
                        if ord(pole2[m])=ord('<') then
                            if ord(pole2[m+1])=ord('c') then
                                if ord(pole2[m+2])=ord('a') then
                                    if ord(pole2[m+3])=ord('s') then
                                        if ord(pole2[m+4])=ord('>') then

                                            begin

hodA:=hodA+(10*StrToInt (pole2 [m+5])+StrToInt (pole2 [m+6])) *ks;

minA:=minA+(10*StrToInt (pole2 [m+8])+StrToInt (pole2 [m+9])) *ks;

secA:=secA+(10*StrToInt (pole2 [m+11])+StrToInt (pole2 [m+12])) *ks;

secAA:=(10*3600*StrToInt (pole2 [m+5])+3600*StrToInt (pole2 [m+6])+10*60*StrToI
nt (pole2 [m+8])+60*StrToInt (pole2 [m+9])+10*StrToInt (pole2 [m+11])+StrToInt (po
le2 [m+12])) *ks;

                            end;
                        end;

                    if cov=1 then
                        begin
                            kA:=1;
                            kontrola:=0;
                            repeat
                                if secAA <(PcasA[kA+1]-PcasA[kA]) then
                                    begin
                                        CPO:=PcasA[kA]+secAA;
                                        PcasA[kA]:=PcasA[kA]+secAA;
                                        kA:=(length(PcasA)-1);
                                        inc(kontrola);
                                    end
                                else
                                    kA:=kA+2;
                            until kA=(length(PcasA)-1);
                            if kontrola=0 then
                                begin
                                    casA:=casA+secAA;
                                    CPO:=casA;
                                end;
                            end
                        else
                            begin
                                if casA > CPO then
```

```
begin
  kA:=1;
  kontrola:=0;

  repeat
  if CPO < PcasA[kA+1] then
  begin
  if CPO > PcasA[kA] then
  begin
  if secAA < (PcasA[kA+1]-CPO) then
  begin
  Prvni:=PcasA[kA];
  Posledni:=CPO;
  PcasA[kA]:=CPO+secAA;
  CPO:=CPO+secAA;

  for k:=(length(PcasA)-2) downto kA do
  begin
  PcasA[k+2]:=PcasA[k];
  end;

  PcasA[kA]:=Prvni;
  PcasA[kA+1]:=Posledni;
  kA:=(length(PcasA)-1);
  inc(kontrola);
  end;
  end;
  if CPO = PcasA[kA] then
  begin
  if secAA < (PcasA[kA+1]-CPO) then
  begin
  PcasA[kA]:=CPO+secAA;
  CPO:=CPO+secAA;
  kA:=(length(PcasA)-1);
  inc(kontrola);
  end;
  end;
  if CPO < PcasA[kA] then
  begin
  if secAA < (PcasA[kA+1]-PcasA[kA]) then
  begin
  CPO:=PcasA[kA]+secAA;
  PcasA[kA]:=PcasA[kA]+secAA;
  kA:=(length(PcasA)-1);
  inc(kontrola);
  end
  end;
  end;
  if kontrola=0 then kA:=kA+2;
  until kA=(length(PcasA)-1);

  if kontrola=0 then
  begin
  casA:=casA+secAA;
  CPO:=casA;
  end;
  end;
  if casA < CPO then
  begin
  kA:=0;
```



```
        repeat
            inc(kA);
        until PcasA[kA]=-1;
        PcasA[kA]:=casA;
        PcasA[kA+1]:=CPO;
        casA:=CPO+secAA;
        CPO:=casA;
    end;
    if casA = CPO then
        begin
            CPO:=casA+secAA;
            casA:=casA+secAA;
        end;
    end;
end;
//B
if ord(pole2[o+12])=ord('B') then
begin

    for m:=o to (o+20) do

        begin
            if ord(pole2[m])=ord('<') then
            if ord(pole2[m+1])=ord('c') then
            if ord(pole2[m+2])=ord('a') then
            if ord(pole2[m+3])=ord('s') then
            if ord(pole2[m+4])=ord('>') then

                begin
hodB:=hodB+(10*StrToInt(pole2[m+5])+StrToInt(pole2[m+6]))*ks;

minB:=minB+(10*StrToInt(pole2[m+8])+StrToInt(pole2[m+9]))*ks;

secB:=secB+(10*StrToInt(pole2[m+11])+StrToInt(pole2[m+12]))*ks;
                secBB:=casvsec(pole2,m,ks);
                end;
            end;
        if cov=1 then
        begin
            kA:=1;
            kontrola:=0;
            repeat
                if secBB < (PcasB[kA+1]-PcasB[kA]) then
                begin
                    CPO:=PcasB[kA]+secBB;
                    PcasB[kA]:=PcasB[kA]+secBB;
                    kA:=(length(PcasB)-1);
                    inc(kontrola);
                end
                else
                    kA:=kA+2;
            until kA=(length(PcasB)-1);
            if kontrola=0 then
            begin
                casB:=casB+secBB;
                CPO:=casB;
            end;
        end
        else
        begin
```

```
if casB > CPO then
begin
  kA:=1;
  kontrola:=0;

  repeat
  if CPO < PcasB[kA+1] then
  begin
    if CPO > PcasB[kA] then
    begin
      if secBB < (PcasB[kA+1]-CPO) then
      begin
        Prvni:=PcasB[kA];
        Posledni:=CPO;
        PcasB[kA]:=CPO+secBB;
        CPO:=CPO+secBB;

        for k:=(length(PcasB)-2) downto kA do
        begin
          PcasB[k+2]:=PcasB[k];
        end;

        PcasB[kA]:=Prvni;
        PcasB[kA+1]:=Posledni;
        kA:=(length(PcasB)-1);
        inc(kontrola);
      end;
    end;
    if CPO = PcasB[kA] then
    begin
      if secBB < (PcasB[kA+1]-CPO) then
      begin
        PcasB[kA]:=CPO+secBB;
        CPO:=CPO+secBB;
        kA:=(length(PcasB)-1);
        inc(kontrola);
      end;
    end;
    if CPO < PcasB[kA] then
    begin
      if secBB < (PcasB[kA+1]-PcasB[kA]) then
      begin
        CPO:=PcasB[kA]+secBB;
        PcasB[kA]:=PcasB[kA]+secBB;
        kA:=(length(PcasB)-1);
        inc(kontrola);
      end;
    end;
    if kontrola=0 then kA:=kA+2;
  until kA=(length(PcasB)-1);
  if kontrola=0 then
  begin
    casB:=casB+secBB;
    CPO:=casB;
  end;
end
else begin
  kA:=0;
```

```
        repeat
            inc(kA);
        until PcasB[kA]=-1;
        PcasB[kA]:=casB;
        PcasB[kA+1]:=CPO;
        casB:=CPO+secBB;
        CPO:=casB;
    end;
end;
end;
//C
if ord(pole2[o+12])=ord('C') then
begin

    for m:=o to (o+20) do

        begin
            if ord(pole2[m])=ord('<') then
            if ord(pole2[m+1])=ord('c') then
            if ord(pole2[m+2])=ord('a') then
            if ord(pole2[m+3])=ord('s') then
            if ord(pole2[m+4])=ord('>') then
            begin
hodC:=hodC+(10*StrToInt(pole2[m+5])+StrToInt(pole2[m+6]))*ks;

minC:=minC+(10*StrToInt(pole2[m+8])+StrToInt(pole2[m+9]))*ks;

secC:=secC+(10*StrToInt(pole2[m+11])+StrToInt(pole2[m+12]))*ks;
                secCC:=casvsec(pole2,m,ks);
            end;
        end;
    if cov=1 then
    begin
        kA:=1;
        kontrola:=0;
        repeat
            if secCC < (PcasC[kA+1]-PcasC[kA]) then
            begin
                PcasC[kA]:=PcasC[kA]+secCC;
                CPO:=PcasC[kA];
                kA:=(length(PcasC)-1);
                inc(kontrola);
            end
            else
                kA:=kA+2;
            until kA=(length(PcasC)-1);
            if kontrola=0 then
            begin
                casC:=casC+secCC;
                CPO:=casC;
            end;
        end
    else
    begin
        if casC > CPO then
        begin
            kA:=1;
            kontrola:=0;
            repeat
                if CPO < PcasC[kA+1] then
```

```
begin
if CPO > PcasC[kA] then
begin
if secCC < (PcasC[kA+1]-CPO) then
begin
Prvni:=PcasC[kA];
Posledni:=CPO;
PcasC[kA]:=CPO+secCC;
CPO:=CPO+secCC;

for k:=(length(PcasC)-2) downto kA do
begin
PcasC[k+2]:=PcasC[k];
end;
PcasC[kA]:=Prvni;
PcasC[kA+1]:=Posledni;
kA:=(length(PcasC)-1);
inc(kontrola);
end;
end;
if CPO = PcasC[kA] then
begin
if secCC < (PcasC[kA+1]-CPO) then
begin
PcasC[kA]:=CPO+secCC;
CPO:=CPO+secCC;
kA:=(length(PcasC)-1);
inc(kontrola);
end;
end;
if CPO < PcasC[kA] then
begin
if secCC < (PcasC[kA+1]-PcasC[kA]) then
begin
CPO:=PcasC[kA]+secCC;
PcasC[kA]:=PcasC[kA]+secCC;
kA:=(length(PcasC)-1);
inc(kontrola);
end
end;
end;
if kontrola=0 then kA:=kA+2;
until kA=(length(PcasC)-1);

if kontrola=0 then
begin
casC:=casC+secCC;
CPO:=casC;
end;
end
else begin
kA:=0;
repeat
inc(kA);
until PcasC[kA]=-1;
PcasC[kA]:=casC;
PcasC[kA+1]:=CPO;
casC:=CPO+secCC;
CPO:=casC;
end;
end;
```



```
        for k:=(length(PcasD)-2) downto kA do
            begin
                PcasD[k+2]:=PcasD[k];
            end;
        PcasD[kA]:=Prvni;
        PcasD[kA+1]:=Posledni;
        kA:=(length(PcasD)-1);
        inc(kontrola);
    end;
end;
if CPO = PcasD[kA] then
begin
if secDD < (PcasD[kA+1]-CPO) then
begin
PcasD[kA]:=CPO+secDD;
CPO:=CPO+secDD;
kA:=(length(PcasD)-1);
inc(kontrola);
end;
end;
if CPO < PcasD[kA] then
begin
if secDD < (PcasD[kA+1]-PcasD[kA]) then
begin
CPO:=PcasD[kA]+secDD;
PcasD[kA]:=PcasD[kA]+secDD;
kA:=(length(PcasD)-1);
inc(kontrola);
end
end;
end;
if kontrola=0 then kA:=kA+2;
until kA=(length(PcasD)-1);

if kontrola=0 then
begin
casD:=casD+secDD;
CPO:=casD;
end;
end
else begin
kA:=0;
repeat
inc(kA);
until PcasD[kA]=-1;
PcasD[kA]:=casD;
PcasD[kA+1]:=CPO;
casD:=CPO+secDD;
CPO:=casD;
end;
end;
end;
//E
if ord(pole2[o+12])=ord('E') then
begin

for m:=o to (o+20) do
begin
if ord(pole2[m])=ord('<') then
```

Katedra průmyslového inženýrství a managementu

Jan Průcha

```
        if ord(pole2[m+1])=ord('c') then
        if ord(pole2[m+2])=ord('a') then
        if ord(pole2[m+3])=ord('s') then
        if ord(pole2[m+4])=ord('>') then

            begin
hodE:=hodE+(10*StrToInt (pole2[m+5])+StrToInt (pole2[m+6])) *ks;

minE:=minE+(10*StrToInt (pole2[m+8])+StrToInt (pole2[m+9])) *ks;

secE:=secE+(10*StrToInt (pole2[m+11])+StrToInt (pole2[m+12])) *ks;
        secEE:=casvsec (pole2,m,ks) ;
        end;
        end;
        if cov=1 then
        begin
        kA:=1;
        kontrola:=0;
        repeat
            if secEE < (Pcase[kA+1]-Pcase[kA]) then
            begin
            Pcase[kA] :=Pcase[kA]+secEE;
            CPO:=Pcase[kA];
            kA:=(length (Pcase)-1) ;
            inc(kontrola);
            end
            else
            kA:=kA+2;
        until kA=(length (Pcase)-1) ;
        if kontrola=0 then
            begin
            casneE:=casneE+secEE;
            CPO:=casneE;
            end;
        end
        else
        begin
        if casneE > CPO then
        begin
        kA:=1;
        kontrola:=0;
        repeat
            if CPO < Pcase[kA+1] then
            begin
            if CPO > Pcase[kA] then
            begin
            if secEE < (Pcase[kA+1]-CPO) then
            begin
            Prvni:=Pcase[kA];
            Posledni:=CPO;
            Pcase[kA] :=CPO+secEE;
            CPO:=CPO+secEE;

            for k:=(length (Pcase)-2) downto kA do
            begin
            Pcase[k+2] :=Pcase[k];
            end;
            Pcase[kA] :=Prvni;
            Pcase[kA+1] :=Posledni;
            kA:=(length (Pcase)-1) ;
```

```
        inc(kontrola);
        end;
    end;
    if CPO = Pcase[kA] then
    begin
    if secEE < (Pcase[kA+1]-CPO) then
    begin
        Pcase[kA]:=CPO+secEE;
        CPO:=CPO+secEE;
        kA:=(length(Pcase)-1);
        inc(kontrola);
        end;
    end;
    if CPO < Pcase[kA] then
    begin
    if secEE < (Pcase[kA+1]-Pcase[kA]) then
    begin
        CPO:=Pcase[kA]+secEE;
        Pcase[kA]:=Pcase[kA]+secEE;
        kA:=(length(Pcase)-1);
        inc(kontrola);
        end
    end;
    if kontrola=0 then kA:=kA+2;
    until kA=(length(Pcase)-1);
    if kontrola=0 then
    begin
        casneE:=casneE+secEE;
        CPO:=casneE;
        end;
    end
    else begin
        kA:=0;
        repeat
            inc(kA);
        until Pcase[kA]=-1;
        Pcase[kA]:=casneE;
        Pcase[kA+1]:=CPO;
        casneE:=CPO+secEE;
        CPO:=casneE;
        end;
    end;
    end;
    end;
    end;
    inc(o);
    end;
    end;
    inc(j);
    end;
    end;
    end;

secA:=3600*hodA+60*minA+secA;
    hodA:=secA div 3600;
    minA:=secA mod 3600;
    secA:=minA mod 60;
    minA:=minA div 60;

secB:=3600*hodB+60*minB+secB;
```


Katedra průmyslového inženýrství a managementu

Jan Průcha

```
    hodB:=secB div 3600;
    minB:=secB mod 3600;
    secB:=minB mod 60;
    minB:=minB div 60;

secC:=3600*hodC+60*minC+secC;
    hodC:=secC div 3600;
    minC:=secC mod 3600;
    secC:=minC mod 60;
    minC:=minC div 60;

secD:=3600*hodD+60*minD+secD;
    hodD:=secD div 3600;
    minD:=secD mod 3600;
    secD:=minD mod 60;
    minD:=minD div 60;

secE:=3600*hodE+60*minE+secE;
    hodE:=secE div 3600;
    minE:=secE mod 3600;
    secE:=minE mod 60;
    minE:=minE div 60;

hodinky(hodA,minA,secA,memoA);
hodinky(hodB,minB,secB,memoB);
hodinky(hodC,minC,secC,memoC);
hodinky(hodD,minD,secD,memoD);
hodinky(hodE,minE,secE,memoE);
memoA.text:='Pracoviště A '#13#10+memoA.text;
memoB.text:='Pracoviště B '#13#10+memoB.text;
memoC.text:='Pracoviště C '#13#10+memoC.text;
memoD.text:='Pracoviště D '#13#10+memoD.text;
memoE.text:='Pracoviště E '#13#10+memoE.text;

    konst:=StrToInt(editKonst.text);
    MemoA.width:=(10*(secA+minA*60+hodA*3600))div (konst);
    MemoB.width:=(10*(secB+minB*60+hodB*3600))div (konst);
    MemoC.width:=(10*(secC+minC*60+hodC*3600))div (konst);
    MemoD.width:=(10*(secD+minD*60+hodD*3600))div (konst);
    MemoE.width:=(10*(secE+minE*60+hodE*3600))div (konst);

    if MemoA.Width+MemoB.Width+MemoC.Width+MemoD.Width+MemoE.Width=0 then
showmessage('Není načtený platný objednávkový list nebo není nastavené
správné měřítko velikosti.');
```

```
    sirkamemicka(casA,konst,MemoAA);
    sirkamemicka(casB,konst,MemoBB);
    sirkamemicka(casC,konst,MemoCC);
    sirkamemicka(casD,konst,MemoDD);
    sirkamemicka(casneE,konst,MemoEE);

    Ahod:=casA div 3600;
    Amin:=casA mod 3600;
    Asec:=Amin mod 60;
    Amin:=Amin div 60;

    Bhod:=casB div 3600;
    Bmin:=casB mod 3600;
    Bsec:=Bmin mod 60;
```

Katedra průmyslového inženýrství a managementu

Jan Průcha

```
Bmin:=Bmin div 60;

Chod:=casC div 3600;
Cmin:=casC mod 3600;
Csec:=Cmin mod 60;
Cmin:=Cmin div 60;

Dhod:=casD div 3600;
Dmin:=casD mod 3600;
Dsec:=Dmin mod 60;
Dmin:=Dmin div 60;

Ehod:=casneE div 3600;
Emin:=casneE mod 3600;
Esec:=Emin mod 60;
Emin:=Emin div 60;

memoAA.Text:=IntToStr(Ahod)+' ':'+IntToStr(Amin)+' ':'+IntToStr(Asec);

hodinky(Ahod,Amin,Asec,memoAA);
hodinky(Bhod,Bmin,Bsec,memoBB);
hodinky(Chod,Cmin,Csec,memoCC);
hodinky(Dhod,Dmin,Dsec,memoDD);
hodinky(Ehod,Emin,Esec,memoEE);
memicko.Text:='cas A: ' + IntToStr(casA) +'÷ cas B: ' + IntToStr(casB)
+'÷ cas C: ' + IntToStr(casC) +'÷ cas D: ' + IntToStr(casD) +'÷ cas E: '
+ IntToStr(casneE);

memoAA.text:='Pracoviště A '#13#10+memoAA.text;
memoBB.text:='Pracoviště B '#13#10+memoBB.text;
memoCC.text:='Pracoviště C '#13#10+memoCC.text;
memoDD.text:='Pracoviště D '#13#10+memoDD.text;
memoEE.text:='Pracoviště E '#13#10+memoEE.text;
if LabelMeritko.Visible = true then LabelMeritko.Visible:=false;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
close;
end;
end.
```

Příloha 2

Objednávkový list

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
```

```
<Objednaci_list>
  <Objednavka Zakaznik="StrakaL">
    <name>Horske kolo Freiman</name>
    <cena>15.375Kc</cena>
    <symbol>HK_Frm</symbol>
    <pocet>3</pocet>
    <Sestava>
      <Podsestava1>Predni vidlice
        <Kolo>
          <znak>K55G</znak>
          <vyroba>
            <pracoviste A>
              <cas>00:02:27</cas>
            </pracoviste A>
            <pracoviste B>
              <cas>00:01:12</cas>
            </pracoviste B>
          </vyroba>
        </Kolo>
        <Vidlice>
          <znak>Vid5536Al</znak>
          <vyroba>
            <pracoviste A>
              <cas>00:01:53</cas>
            </pracoviste A>
            <pracoviste C>
              <cas>00:21:18</cas>
            </pracoviste C>
          </vyroba>
        </Vidlice>
        <Ram>
          <znak>RamVid5520Al</znak>
          <vyroba>
            <pracoviste D>
              <cas>00:12:18</cas>
            </pracoviste D>
            <pracoviste C>
              <cas>00:12:36</cas>
            </pracoviste c>
            <pracoviste B>
              <cas>00:07:36</cas>
            </pracoviste B>
          </vyroba>
        </Ram>
        <OsaPredni>
          <znak>OsaP3620Fe</znak>
          <vyroba>
            <pracoviste A>
              <cas>00:07:08</cas>
            </pracoviste A>
            <pracoviste C>
```

```
<cas>00:10:56</cas>
</pracoviste c>
<pracoviste B>
<cas>00:02:36</cas>
</pracoviste B>
</vyroba>
</OsaPredni>
</Podsestava1>
<Podsestava2>Zadni vidlice
<Kolo>
<znak>K55G</znak>
<vyroba>
<pracoviste A>
<cas>00:02:27</cas>
</pracoviste A>
<pracoviste B>
<cas>00:01:12</cas>
</pracoviste B>
</vyroba>
</Kolo>
<ZadniVidlice>
<znak>Vid5536A1</znak>
<vyroba>
<pracoviste A>
<cas>00:12:17</cas>
</pracoviste A>
<pracoviste A>
<cas>00:03:12</cas>
</pracoviste A>
<pracoviste C>
<cas>00:21:18</cas>
</pracoviste C>
</vyroba>
</ZadniVidlice>
<Ram>
<znak>RamVid5520A1</znak>
<vyroba>
<pracoviste D>
<cas>00:12:18</cas>
</pracoviste D>
<pracoviste C>
<cas>00:12:36</cas>
</pracoviste c>
<pracoviste B>
<cas>00:07:36</cas>
</pracoviste B>
</vyroba>
</Ram>
<OsaZadni>
<znak>OsaZ3620Fe</znak>
<vyroba>
<pracoviste A>
<cas>00:18:22</cas>
```

```
</pracoviste A>
<pracoviste C>
  <cas>00:17:51</cas>
</pracoviste C>
<pracoviste E>
  <cas>01:03:48</cas>
</pracoviste E>
<pracoviste B>
  <cas>00:17:28</cas>
</pracoviste B>
</vyroba>
</OsaZadni>
<Montaz>
  <znak>OsaZ3620Fe</znak>
  <vyroba>
    <pracoviste E>
      <cas>00:23:48</cas>
    </pracoviste E>
  </vyroba>
</Montaz>
</Podsestava2>

</Sestava>
</Objednavka>
<Objednavka Zakaznik="KulikR">
  <name>Kolo</name>
  <cena>1675Kc</cena>
  <symbol>K55G</symbol>
  <pocet>2</pocet>
  <Sestava>
    <Kolo>
      <znak>K55G</znak>
      <vyroba>
        <pracoviste A>
          <cas>00:02:27</cas>
        </pracoviste A>
        <pracoviste B>
          <cas>00:01:12</cas>
        </pracoviste B>
      </vyroba>
    </Kolo>
  </Sestava>
</Objednavka>

<Objednavka Zakaznik="MorelA">
  <name>Kolo</name>
  <cena>1675Kc</cena>
  <symbol>K55G</symbol>
  <pocet>1</pocet>
  <Sestava>
    <Kolo>
      <znak>K55G</znak>
      <vyroba>
        <pracoviste A>
```

Katedra průmyslového inženýrství a managementu

Jan Průcha

```
<cas>00:02:27</cas>
</pracoviste A>
<pracoviste B>
<cas>00:01:12</cas>
</pracoviste B>
</vyroba>
</Sestava>
</Objednavka>

</Objednaci_list>
```