

**VŠB - Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**

**Návrh univerzálního interface mezi systémy ERP (Enterprise Resource Planning)  
a WMS (Warehouse Management System)**  
**Design of Universal Interface between ERP (Enterprise Resource Planning) and  
WMS (Warehouse Management System)**

## Zadání diplomové práce

Student: **Bc. Vladimír Vašek**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2612T025 Informatika a výpočetní technika

Téma: **Navrh univerzálního interface mezi systémy ERP (Enterprise Resource Planning) a WMS (Warehouse Management System)  
Design of Universal Interface between ERP (Enterprise Resource Planning) and WMS (Warehouse Management System)**

### Zásady pro vypracování:

Vytvořit návrh prakticky použitelného univerzálního interface mezi podnikovými informačními systémy (ERP) a systémy pro řízení skladových operací (WMS). Navržený interface musí být univerzálně použitelný pro jakýkoliv ERP a WMS systém bez ohledu na použitou platformu.

1. Definujte typické průnikové požadavky ERP a WMS a požadavky na jejich integraci.
2. Analyzujte obecné předpoklady pro návrh interface s využitím dostupné odborné literatury.
3. Navrhnete obecné rozhraní - seznam případů užití, strukturu vyměňovaných dat a základní funkčnost operací.
4. Realizujte praktický příklad ověřující použitelnost na vybraném ERP a WMS.
5. Definujte rizika a limity použitelnosti univerzálního interface.

### Seznam doporučené odborné literatury:

Podle pokynů vedoucího diplomové práce.

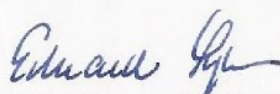
Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Zdeněk Opršal**

Konzultant diplomové práce: Ing. Jan Kožusznik, Ph.D.

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015



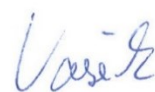
doc. Dr. Ing. Eduard Sojka  
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

## **Prohlášení**

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.



V Ostravě dne 4.5.2015

Vladimír Vašek

## **Poděkování**

Tímto bych chtěl poděkovat Ing. Zdeňkovi Opršalovi a Ing. Janu Kožusznikovi Ph.D. za odborné vedení při tvorbě této diplomové práce.

## **Abstrakt**

Tato diplomová práce se zabývá návrhem rozhraní mezi informačními systémy ERP a WMS. Cílem práce je navrhnout integrační aplikaci zajišťující komunikaci a přenos dat mezi systémy ERP a WMS. První část této práce představuje podnikové informační systémy ERP a systémy pro řízení skladů WMS. Druhá část práce popisuje vlastnosti integrace dat, strukturu middleware a existující řešení poskytující integraci dat mezi systémy. Třetí část této práce se věnuje analýze skladových operací a jejich průběhu, které jsou společné pro systémy ERP a WMS. V této části je také naznačena komunikace mezi integrovanými systémy. Poslední část této diplomové práce se zabývá samotným návrhem a implementací integrační aplikace. Popisuje použité technologie, návrh tříd a vyřízení požadavku skladové operace.

## **Abstract**

This master thesis describes the design of the interface between ERP and WMS information systems. The aim is to design an integration application providing communication and data transfer between ERP and WMS. The first part presents ERP enterprise information systems and WMS storage systems. The second part describes the characteristics of data integration, middleware structure and existing solutions providing data integration between systems. The third part of this thesis provides an analysis of warehouse operations and their workflow, which are common for ERP and WMS. In this section is also indicated communication between integrated systems. The last part of this thesis deals with the design and implementation of integration application. It describes the technology, design and processing of requirement operations from warehouse.

## **Klíčová slova**

Podnikové informační systémy, ERP, WMS, middleware, integrační aplikace, WCF, webové služby

## **Key Words**

Enterprise information system, ERP, WMS, middleware, integration application, WCF, web services

## Seznam použitých zkratk a symbolů

ACL	-	Access Control List
APS	-	Advanced Planning and Scheduling
BAM	-	Business Activity Monitoring
BEM	-	Business Event Management
BI	-	Business Intelligence
BPM	-	Business Process Management
BRE	-	Business Rules Engine
CORBA	-	Common Object Request Broker Architecture
CRM	-	Customer Relationship Management
DCOM	-	Distributed Component Object Model
EDI	-	Electronic Data Interchange
ERP	-	Enterprise Resource Planning
ESB	-	Enterprise Service Bus
ETL	-	Extract, Transform and Load
FIFO	-	First In First Out
FTP	-	File Transfer Protocol
HTTP	-	Hypertext Transfer Protocol
IIS	-	Internet Information Service
JDBC	-	Java Database Connectivity
JMS	-	Java Message Service
JVM	-	Java Virtual Machine
MOM	-	Message-oriented Middleware
MS	-	Microsoft
MSMQ	-	Microsoft Message Queuing
ODBC	-	Open Database Connectivity
OLE	-	Object Linking and Embedding
OLE DB	-	Object Linking and Embedding, Database
PIP	-	Packard Integrating Process
RFID	-	Radio Frequency Identification
RMI	-	Remote Method Invocation
RPC	-	Remote Procedure Call
SOA	-	Service Oriented Architecture
SOAP	-	Simple Object Access Protocol
SQL	-	Structured Query Language
SSIS	-	SQL Server Integration Server
SSO	-	Single Sign On
TCP/IP	-	Transmission Control Protocol/Internet Protocol
UC	-	Use Case

- UML - Unified Modeling Language
- W3C - World Wide Web Consortium
- WCF - Windows Communication Foundation
- WMS - Warehouse Management System
- WSDL - Web Services Description Language
- WSE - Web Services Enhancements
- XML - Extensible Markup Language

# Obsah

1	Úvod.....	1
2	Informační systémy organizací .....	2
2.1	Stavba a funkce ERP systémů.....	3
2.1.1	Finanční management .....	5
2.1.2	Prodej, marketing a služby .....	5
2.1.3	Řízení dodavatelského řetězce .....	5
2.1.4	Business Intelligence.....	5
2.2	Struktura dat v systémech ERP .....	5
2.3	Funkcionalita ERP systémů .....	6
2.4	Systémy pro řízení skladových operací.....	6
2.4.1	Přínosy systémů pro řízené sklady .....	7
3	Integrace dat mezi informačními systémy ERP a WMS .....	9
3.1	Funkcionalita middleware .....	9
3.1.1	Funkcionalita základního middleware.....	10
3.1.2	Funkcionalita integračního middleware .....	11
3.1.3	Funkcionalita middleware pro aplikační integraci .....	12
3.2	Existující integrační nástroje.....	12
3.2.1	SQL Server Integration Server.....	13
3.2.2	BizTalk .....	14
3.2.3	SonicMQ .....	15
3.2.4	Cordys SOA Grid.....	16
3.3	Volba technologie pro integraci ERP a WMS systému.....	17
3.3.1	Zprávy ve webových službách .....	18
3.3.2	Dokument popisující webovou službu .....	19
4	Analýza integrační aplikace .....	21
4.1	Definice pojmů užívaných při skladových operacích .....	21
4.2	Jednotlivé případy užití a komunikace mezi systémy .....	23
4.2.1	Příjem zboží.....	24
4.2.2	Výdej zboží .....	27

4.2.3	Inventura.....	30
4.2.4	Další skladové operace.....	33
5	Implementace .....	35
5.1	Použité technologie .....	35
5.1.1	Windows Communication Foundation.....	35
5.1.2	Aplikační rámec .NET.....	37
5.1.3	Aplikační rámec Entity Framework .....	38
5.1.4	Návrhový vzor Repository .....	39
5.2	Třídní diagram integrační aplikace.....	39
5.3	Popis implementace zpracování požadavku skladové operace .....	42
5.4	Implementace klienta webové služby.....	46
5.5	Rizika a limity navrženého rozhraní .....	48
6	Závěr .....	50
7	Použitá literatura .....	52
8	Přílohy.....	54



# 1 Úvod

Podnikové informační systémy jsou dnes samozřejmou součástí podnikové infrastruktury u velkých a u většiny středních společností. Zatímco v dřívějších dobách byly tyto podnikové informační systémy určené právě pro velké společnosti kvůli vysokým vstupním nákladům způsobeným rozsáhlostí systémů, dnes se vývojová oddělení soustředí na vývoj podnikových systémů i pro střední a malé společnosti. Společnosti se snaží využívat všech výhod, které jim moderní podnikové systémy nabízejí, avšak vždy je při zavádění takto rozsáhlých systémů potřeba kompromisů ze strany dodavatele systému i ze strany zákazníka. Snahou dodavatelů podnikových systémů je přizpůsobit tyto podnikové informační systémy procesům ve společnosti, ale ne vždy je možné vyhovět všem požadavkům zákazníka. Někdy je nutné upravit i některé zažité procesy společnosti, což bývá problémem právě při novém zavádění podnikových systémů.

Ačkoliv se moderní podnikové informační systémy snaží pokrýt co nejširší spektrum podnikových procesů, někdy je nutné využití specializovaných systémů například pro skladové řízení apod. Důležitou součástí těchto systémů je jejich vzájemná integrace a komunikace. Cílem této práce je analýza skladových operací a navržení univerzálního rozhraní mezi systémy ERP (podnikový informační systém) a systémy WMS (systém pro řízení skladu).

První část této práce se zabývá představením ERP systémů, jejich vlastnostmi, architekturou a funkcemi. Dále popisuje hlavní strukturu dat, kterou tyto systémy poskytují a jež jsou jejich součástí. Následuje stručné představení systémů WMS a jejich hlavních vlastností a výhod.

Druhá část této práce se zaměřuje na potřeby integrace těchto systémů, popisuje jednotlivé postupy při integraci a existující řešení poskytující integraci dat mezi jednotlivými podnikovými systémy. Dále popisuje architekturu a vybrané návrhové vzory pro middlewarové aplikace umožňující integraci systémů.

Ve třetí části této práce jsou podrobněji popsány a specifikovány procesy potřebné pro integraci systémů ERP a WMS. Jsou zde popsány jednotlivé procesy týkající se skladového hospodářství a průběh komunikace mezi systémy.

Poslední kapitola se zabývá samotným návrhem integrační aplikace. Popisuje zvolenou technologii pro integraci a princip fungování komunikace mezi systémy. Dále popisuje potřebné předpoklady pro implementaci této integrační aplikace do systémů.

## 2 Informační systémy organizací

Informační systém organizace je komplexní systém firemních agend, jako například daňová evidence, účetnictví, fakturace, objednávky, skladová evidence, mzdy, kniha jízd, adresář, výroba atd. Jednotlivým agendám se pak říká moduly. Jako širší součást informačního systému organizace lze chápat také firemní intranet, extranet a různé e-commerce aplikace, které jsou součástí e-businessu [11].

Informační systémy ERP (Enterprise Resource Planning), které integrují a automatizují velké množství procesů ve firmě včetně výroby, logistiky, distribuce, správy majetku, prodeje, fakturace a účetnictví, mají dvě důležité vlastnosti. První z těchto vlastností je integrace dat a druhou je podpora osvědčených postupů. Integrací dat se rozumí, že ihned po zadání dat do systému jsou tato data k dispozici pro použití v rámci celé organizace. Většina organizací má před implementací ERP systému několik paralelních správních orgánů. V takových situacích je normální, že má v rámci jedné společnosti oddělení marketingu registr zákazníků, sklad registr objednávek a finanční oddělení registr prodejních faktur. Tyto registry mohou být elektronické, ale v některých případech se mohou vyskytovat i v klasických šanonech v papírové podobě. Hlavně v druhém případě často dochází ke zdvojení (neboli redundanci) a nekonzistenci dat.

V ERP systému je možné vytvořit jeden integrovaný registr, který mohou využívat všechny správní orgány najednou. Zaměstnancům z různých oddělení mohou být přidělena práva k zobrazení dat jiných oddělení. Díky tomu je možné kombinovat data o zákazníkovi s prodejními objednávkami, případně fakturami a podobně. Jednotlivá oddělení jsou zodpovědná za aktuálnost a kompletnost dat. ERP systémy pak dokáží na základě změněných údajů automaticky aktualizovat data ve všech souvisejících registrech. Například po odeslání objednávky skladem je vytvořena a vytištěna faktura ve finančním oddělení.

ERP systémy zamezují redundanci dat a zvyšují efektivitu zpracování jednotlivých podnikových procesů díky částečné automatizaci. Nejdůležitějším přínosem integrity dat v ERP systémech spočívá v tom, že nevznikají rozdíly v registrech jednotlivých oddělení a nemusí se dohledávat případné důvody vzniku rozdílných dat.

Pojmem osvědčených postupů (*best-practise*) se rozumí ověřené a přijaté způsoby vykonávání daných činností, které byly zavedeny do mnoha organizací, a byl dokázán jejich přínos daným společností. Příkladem pro osvědčený postup může být maximální úvěr poskytovaný zákazníkovi, který snižuje riziko špatného splácení půjčky zákazníkem. Pokud je vytvořena nová zákaznická objednávka, v systému ERP se kontroluje, zdali celková částka nezaplacených faktur dohromady s částkou z předcházejících objednávek a s aktuální částkou nepřekročí maximální definovanou výši úvěru. Maximální úvěr lze takto nastavit každému zákazníkovi a při jakékoli nové objednávce je kontrolováno nepřekročení maximálního úvěru zákazníka. [1]

Existuje velké množství takových osvědčených postupů a nové jsou stále přidávány spolu s rozšiřujícími se ERP systémy. Pokud má společnost nějaké vlastní osvědčené postupy, implementují se při zavádění ERP systému do společnosti.

Systémy ERP jsou schopny podporovat všechny procesy související s podnikovou ekonomikou dané společnosti. Výhody integrace jsou patrné především v „přímém zúčtování“ veškerých obchodních případů ve všech aktivovaných komponentách standardního systému. Pouze díky všeobecnému a úplnému propojení více aplikačních komponent s jedním obchodním procesem je možné ustoupit od využívání různých rozhraní a data zadávat pouze jednou na místě jejich vzniku. Systémy ERP obvykle podporují funkce nezbytné k operativnímu zpracování všech pravidelně se opakujících obchodních případů v daném podniku [12].

## 2.1 Stavba a funkce ERP systémů

Standardní integrované systémy jsou založeny na jednotném vývojovém konceptu. V některých případech nelze ojedinelé a vzájemně nezávisle koncipované funkce zahrnout do celkového systému. Vývojový koncept má obvykle podobu vrstevného modelu. Systémy ERP nejsou určeny k nasazení na jedno pracoviště, podporují totiž takové funkce podniku, které jsou využívány více pracovníky nejen v různých odděleních, ale i v různých lokalitách. Z tohoto důvodu je použití vrstevné architektury nezbytné. Základem této architektury je ve většině případů architektura klient / server oddělující vrstvu databáze od aplikace a prezentace. [12]

V prvních ERP systémech se využívalo nejčastěji dvouvrstvé architektury, kde jednu vrstvu tvořil databázový server, druhou vrstvou byl klient, který sloužil pro prezentaci dat a spuštění aplikační logiky. S příchodem nových technologií, a hlavně potřebou zobrazovat data ze serveru na různých klientských zařízeních, jsou novější ERP systémy založeny na třívrstvé architektuře. V třívrstvé architektuře přibyla mezi vrstvu klienta a databázovou vrstvu ještě vrstva starající se o spuštění logiky. [1]

Základem podnikových informačních systémů je jedna společná databáze, díky níž je možné jednotlivá oddělení podniku propojit. Tím je zaručena aktuálnost dat napříč celým informačním systémem a jednotlivá data lze zadávat pouze jednou při jejich vzniku.

ERP systémy obvykle pracují transakčně, tj. vytvářejí řadu transakcí sloužících k podpoře obchodních procesů. Pojmem transakce se rozumí logicky ukončené operace sestávající z jednotlivých akcí. Na pozadí tak dochází k převodu databáze z původního do nového konzistentního stavu. Z toho vyplývá, že provádění každé akce (transakce) musí být zcela dokončeno nebo nesmí být vůbec zahájeno [12].

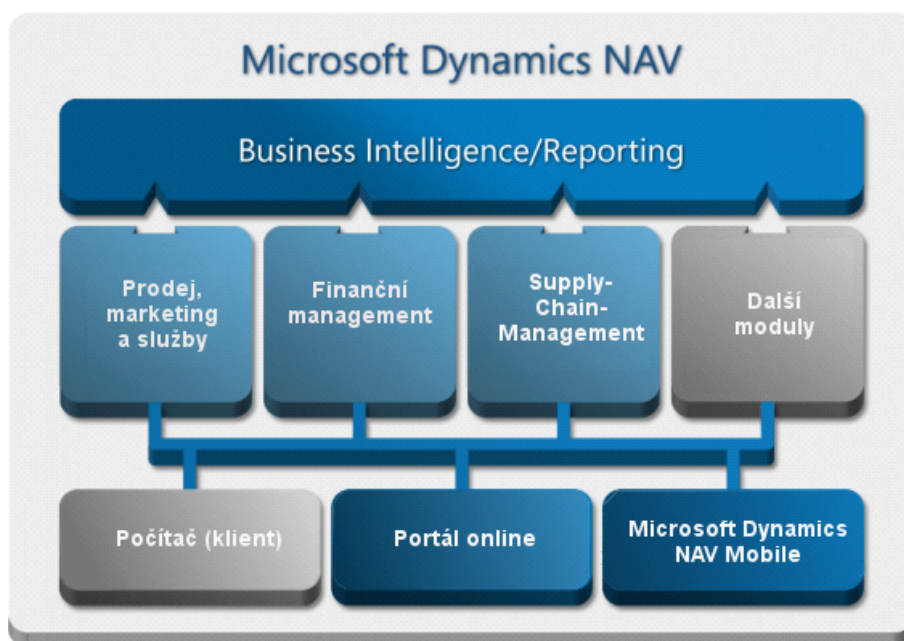
Omezení pro přístup k datům z jiných oblastí se v systémech ERP nezadávají pomocí omezení přes architekturu systému, ale řídí se cíleně příslušnými nastaveními oprávnění. Tím je zajištěna ochrana dat a zároveň jsou všechna potřebná data neustále k dispozici. [1]

Systémy ERP se používají pro zaznamenání, evidenci a následné zobrazení přímých obchodních procesů přes hranice jednotlivých oddělení pomocí sdílení dokladů. Ačkoliv je základní struktura závislá na příslušném oddělení prakticky stejně, existují u středně velkých a velkých podniků většinou i určité speciality týkající se jejich provozu, které poskytují určitou konkurenční výhodu. Proto je nutno systémy ERP nakonfigurovat a upravit tak, aby odpovídaly přesně požadavkům a nárokům každého jednotlivého podniku.

Struktura ERP systémů bývá nejčastěji modulární, přičemž jsou mezi sebou jednotlivé moduly plně integrovány. [1] Modularita ERP systémů poskytuje možnost zakoupení pouze určité části ERP systému, které firma potřebuje.

Podle velikosti firmy a počtu uživatelů se mohou lišit jednotlivé verze ERP systémů od těch nejjednodušších pro malé společnosti, jako je například Money S3 od Cigler Software, přes ERP určené pro střední až velké podniky, například Microsoft Dynamics NAV, až po rozsáhlé ERP systémy, které reprezentuje například nerozšířenější a nejznámější aplikace společnosti SAP, určená pro velké podniky.

Pro příklad modularity ERP systémů je použito schéma jednotlivých modulů pro Microsoft Dynamics NAV. [1]



Obrázek 1 - Struktura MS Dynamics NAV 2013 [1]

Mezi základní moduly, které jsou společné téměř pro všechny ERP systémy, patří finanční management, prodej, marketing a služby, řízení dodavatelského řetězce, business intelligence, lidské zdroje, plánování zdrojů, apod. Nejdůležitější z nich jsou ve stručnosti popsány níže.

### **2.1.1 Finanční management**

Oblast Finančního managementu představuje základ pro většinu ERP systémů, jelikož se díky úzké integraci do tohoto modulu sbíhají transakce ze všech oblastí. Na tento základ pak navazují další rozsáhlé možnosti analýzy a podávání zpráv neboli reporting. Finanční management obsahuje mimo finančního účetnictví a účetnictví podniku i správu závazků, pohledávek, bankovních kont a ocenění skladu. [1]

### **2.1.2 Prodej, marketing a služby**

Oblast prodeje, marketingu a služeb obsahuje funkce pro plánování a řízení aktivit týkajících se marketingu a prodeje. Mezi tyto aktivity patří základní správa kontaktů, zájemců a zákazníků, dále správa aktivit zákazníků a správa kampaní. V oblasti služeb lze cíleně upravovat požadavky zákazníků na služby a dohody o poskytování služeb. [1]

### **2.1.3 Řízení dodavatelského řetězce**

Oblast řízení dodavatelského řetězce (*Supply Chain Managementu*) představuje komplexní řetězec dodávky zahrnující nákup, skladování, výrobu, prodej a logistiku. Vzhledem k tomu, že lze celý dodavatelský řetězec spravovat a plánovat nepřetržitě, je možné minimalizovat dobu běhu a nákladů při současně vyšší připravenosti k dodávce zboží. [1] Tento modul je nejdůležitější z hlediska univerzálního rozhraní mezi systémy ERP a WMS.

### **2.1.4 Business Intelligence**

Tato oblast obsahuje funkce, které se dají použít pro zobrazení centrálních podnikových dat ve snadno srozumitelné podobě. Jedná se hlavně o tabulky a grafy. Prostřednictvím nástroje v tomto modulu je možné velmi snadno vyhodnotit všechny důležité informace týkajících se například počtu včas vyexpedovaných zásilek apod. Díky přímé integraci dat jsou data vždy aktuální a ihned dostupné. [1]

## **2.2 Struktura dat v systémech ERP**

Každé řešení ERP vytváří model organizace a obchodních procesů podniku. V jednotlivých řešeních se rozlišují tři základní typy dat. Prvním typem jsou data, která popisují objekty (například zboží, zákazníci, dodavatele), druhým typem jsou data popisující události neboli transakce v podniku (nákupní objednávka, příjemka na sklad, apod.) a třetím typem jsou data konfigurační. Konfigurační data jsou zapotřebí například pro zobrazování dat a procesů z různých firem.

Konfigurační data se nastavují například při vytváření skladu. Tato data definují například, zda má daná firma sklady na různých místech. Pokud ano, pak je nutné při každém pohybu na skladě

zadat i umístění skladu, v němž k tomuto pohybu došlo. Nutnost vyplnění tohoto pole je uložena právě v konfiguračních datech.

Pomocí hlavních dat se popisují objekty typu zboží, dodavatelé a zákazníci, a mění se jen tehdy, když se změní příslušný objekt. Například když se dodavatel přestěhuje na jinou adresu, jsou upravena odpovídající hlavní data. V oblasti vyhrazené pro základní údaje se samozřejmě často doplňují úplně nové objekty, například nové zboží nebo noví zákazníci a podobně. [1]

Data týkající se transakcí, jako jsou například pohyby na skladu, objednávky pro nákup nebo účtování, popisují operace při vykonávání obchodních operací. Tato data vznikají u každé zakázky, která se vyřizuje. Nejčastěji taková data vznikají zadáním do systému ERP uživatelem nebo při automatizovaném zpracování dokladů systémem ERP. Například při zaúčtování objednávky vzniká automaticky doklad faktury. [1]

## **2.3 Funkcionalita ERP systémů**

Nejnovější ERP systémy podporují všechny řídicí a operační procesy v rámci celé společnosti. Jednotlivé procesy jsou v této společnosti rozděleny do dvou kategorií. První kategorii tvoří primární procesy, které jsou typické pro danou společnost. Mezi takové procesy může patřit například nákup materiálů, výroba, distribuce finálních výrobků, prodej a servis po prodeji. Oproti tomu sekundární procesy nejsou závislé na typu společnosti, ale objevují se ve většině společností. Mezi tyto procesy se řadí zadávání veřejných pohledávek, technologie a údržba, řízení lidských zdrojů, management financí apod.

Jelikož se první ERP systémy zaměřovali na výrobní společnosti, první pokusy o integraci dat se zaměřovaly hlavně na přenos informací mezi primárními a sekundárními procesy, nejvíce se potom zaměřovaly na propojení primárních procesů s finančními procesy.

V osmdesátých letech byly ERP systémy rozšířeny tak, aby již integrovali všechny sekundární procesy. V dnešní době je vysoká poptávka na integraci dat mezi jednotlivými nezávislými autonomně stojícími aplikacemi, které jsou využívány v rámci jedné společnosti i v rámci více organizací. Někdy mohou být například při úzké spolupráci s dodavatelem propojeny systémy daného podniku s některými částmi systému dodavatele.

## **2.4 Systémy pro řízení skladových operací**

Jedná se o označení systémů v distribučních centrech, velkoobchodních skladech, logistických centrech a jiných skladovacích prostorách [1]. Nasazování WMS (systémy pro řízení skladu) do skladů společnosti je vynucováno stále více vzrůstajícím tlakem na zvyšování obrátkovosti zásob, na zvýšení efektivity činností a produktivity práce. Systémy WMS jsou z převážné většiny založeny na bezdrátové komunikaci pomocí Wi-Fi sítě ve skladech a na mobilních terminálech s integrovaným snímačem čárového kódu [1].

Systémy pro řízené sklady nepatří mezi široce používané informační systémy. Systémy WMS nacházejí uplatnění především u těch organizací, které potřebují znát přesnou dispozici svých zásob a efektivněji nakládat s jejich zpracováním a distribucí. [3]

Pod pojmem skladové hospodářství v ERP systémech je míněna účetní a fyzická evidence skladovaného materiálu. Společnost má proto představu o tom, jakými skladovými jednotkami disponuje a v jakém množství. Tyto systémy již ale často nemají detailní údaje například o tom, která část zásoby byla doručena v určitý den, z jakých komponent a polotovarů výrobek vznikl, kam byl uložen, jakým způsobem byl uložen, jak byl expedován apod. [3]

Systém řízených skladů naopak umožňuje detailnější členění skladového hospodářství. Poskytuje velmi podrobnou evidenci jednotlivých položek na úrovni balení, kusů, šarží a skladových pozic. Jeho uživatel pak může zjistit, kde přesně se fyzicky nachází konkrétní balení zboží a kolik dané balení obsahuje kusů. Manipulační obsluha má k dispozici údaje o vazbě na příjmový a výdajový doklad, časech manipulací, příslušnosti k zakázkám, pracovištím výroby, stavech balení (například zablokováno kvalitou) apod. Systém řízených skladů dále nabízí řadu funkcí, které umožňují zjišťovat aktuální stav skladových položek a všech souvisejících atributů. [3,4]

Prvním impulsem, pro rozhodnutí o pořízení systému pro řízení skladů, je potřeba informací o tom, kde se jaký artikl přesně nachází. Je obtížné dohledávat konkrétní balení ve vícepatrovém regálu s cílem vybrat právě to, které aktuálně potřebuje firma zpracovat ve výrobě při respektování různých metodik pohybu zásob tak, aby byla zajištěna plynulá obrátka zásob. K dalším velmi důležitým důvodům pořízení systému řízených skladů patří zvýšení využitelnosti skladové kapacity, zajištění zpětné sledovatelnosti, snížení a zpřesnění stavu zásob, zefektivnění práce skladové obsluhy, snížení počtu reklamací a zrychlení inventurních prací. [3]

Systémy WMS jsou založeny na jednoznačné identifikaci každého zaskladněného materiálu (zboží) čárovým kódem, kterým jsou označeny také pozice všech skladových míst. Pracovníci skladu jsou vybaveni například mobilními terminály založenými na bezdrátové komunikaci s integrovaným snímačem čárového kódu. Na displeji těchto terminálů jsou pro skladníky zobrazovány všechny pokyny a informace k provádění skladových operací, které jsou potvrzovány čtením čárových kódů ze zboží a skladových míst například při vychystávání zboží. [1]

#### **2.4.1 Přínosy systémů pro řízené sklady**

WMS musí být schopen uřídit všechny skladové procesy od příjmu materiálu, zaskladnění, výdeje do výroby až po expedici, kontrolu a inventuru. Dále musí umět zjištěné informace o procesech při řízení skladu přenést do ostatních částí informačních systémů společnosti a oboustranně s nimi komunikovat, např. prostřednictvím EDI. V neposlední řadě musí systém řízených skladů umět zvládnout specifické požadavky zákazníka plynoucí z jeho oborového

zaměření a velikosti. V současné době je samozřejmostí podpora pokročilých technologií, jako je RFID, apod. [4]

Mezi hlavní přínosy WMS patří zvýšení průchodnosti skladu a celkové zrychlení logistických procesů. Systém řízených skladů také umožňuje sledovatelnost a dohledatelnost veškerého materiálu, polotovarů či hotových výrobků a sledování pohybu obalů. Další přínosy úzce souvisejí se zefektivněním práce skladové obsluhy. Jde především o nulovou chybovost při manipulaci (zejména v expedici) a úsporu času. Systém přesně navádí skladníky ke konkrétnímu artiklu, takže při výdeji nemusí nic hledat a přemýšlet nad tím, zda je požadovaná distribuce materiálu či výrobku správná. [3,4]



## 3 Integrace dat mezi informačními systémy ERP a WMS

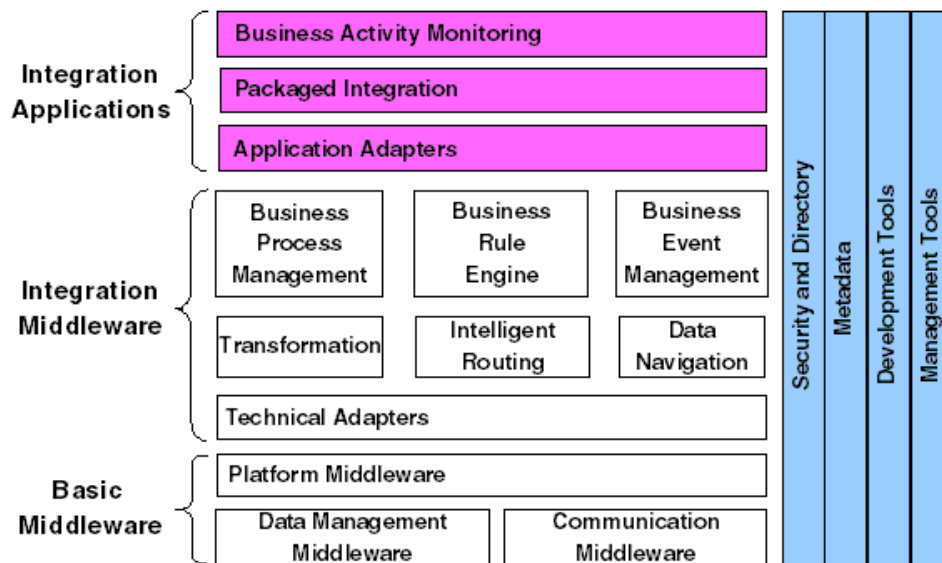
Ačkoliv už v dnešní době systémy ERP obsahují základní funkčnost pro řízení skladů, nejsou vždy zcela dostatečné pro potřeby společnosti a neobsahují funkce, které mohou poskytovat specializované WMS systémy. Z tohoto důvodu často dochází k tomu, že společnost, která již vlastní ERP systém, dokoupí a využívá systém WMS pro správu a řízení skladu. Oba dva systémy mohou pracovat autonomně, ale jejich integrace může být pro společnost velkým přínosem. Nejčastěji se integruje WMS systém do nadřazeného ERP systému, který zpracovává údaje odeslané ze systému WMS a následně provádí další úkony jako například vystavení faktury.

Mezi systémy ERP a WMS je nutná synchronizace dat v reálném čase, aby byla zachována aktuálnost a přesnost dat například o dostupnosti zboží, kde se zboží nachází apod. Naproti tomu systém WMS přijímá zboží na sklad a vydává zboží ze skladu na základě požadavku ze systému ERP. Jelikož se jedná o dva odlišné a samostatné systémy, je nutné zajistit správnou komunikaci mezi těmito systémy pomocí integrace dat.

V dnešní době existuje celá řada služeb postavených na různých platformách, které zprostředkovávají komunikaci mezi jednotlivými systémy. Takové služby se často označují jako middleware. Pojem middleware může být definován mnoha různými způsoby, jelikož se nejedná o ustálený pojem. Pro příklad je zde uvedena definice pojmu middleware od společnosti Gartner: „*Služby vytvářejí provozní systémové prostředí, které umožňuje procesům, programům a aplikacím vzájemnou interakci v distribuovaném systému prostřednictvím sítě*“ [12]. Middleware tedy může být chápán jako aplikace stojící mezi dvěma systémy, která vytváří most pro přenos dat. Velmi často se jedná o webové služby, které nemají žádné uživatelské rozhraní a umožňují komunikaci mezi systémy postavenými na různých platformách.

### 3.1 Funkcionalita middleware

Middleware nástroje lze podle funkcionality rozdělit na základní middleware, integrační middleware a prostředky aplikační integrace. Každý z prostředků nabízí vedle konkrétní funkcionality i služby bezpečnosti, vývoje a správy [2].



Obrázek 2 – Struktura middleware [15]

Na obrázku je znázorněná struktura middleware, která je rozdělena do třech hlavních kategorií. První kategorii tvoří základní middleware, který je následován integračním middleware a poslední kategorii tvoří middleware pro integrační aplikaci.

### 3.1.1 Funkcionalita základního middleware

Z obrázku 2 je patrné rozdělení funkcionality základního middleware do dalších kategorií:

- **komunikační middleware (*Communication Middleware*)** – Je určen pro vzájemnou komunikaci mezi dvěma programy v různých aplikacích nebo i v rámci jedné aplikace. Tato vrstva poskytuje protokol pro přenos dat a zpráv mezi programy a programovým rozhraním, jehož prostřednictvím program přistupuje ke komunikačním službám. Tato vrstva je podporována různými technologiemi, které mohou být založeny na asynchronní i synchronní komunikaci. Příkladem pro synchronní komunikaci mohou být technologie RPC (Remote Procedure Call), RMI (Remote Method Invocation), CORBA (Common Object Request Broker Architecture), DCOM (Distributed Component Object Model) a webové služby, které jsou v dnešním prostředí vysoce používané. Asynchronní komunikaci představují prostředky řízení fronty zpráv (message-oriented middleware). [15]
- **prostředky řízení přístupu k datům (*Data Management Middleware*)** – zajišťuje pomocnou funkcionalitu programům pro přístup k datům, která jsou uložena v různých datových zdrojích, mezi které patří čtení či zápis do vzdálených databází či souborů. Do této oblasti patří různé knihovny a API rozhraní jako je například knihovna ODBC (Open Database Connectivity) a nebo JDBC (Java Database Connectivity). [15]

- **platformový middleware (*Platform middleware*)** - poskytuje vhodné provozní prostředí s množinou obecně použitelných služeb. Zjednodušeně řečeno poskytuje běhové (angl. „runtime“) prostředí a služby pro běh aplikací (např. správa paměti, operační systém, procesy a vlákna, načítání programů z disku dle potřeby, spuštění a zastavování programů, odolnost proti chybám, zabezpečení přístupu, monitoring, distribuované zpracování transakcí atd.). Představitelé této vrstvy jsou aplikační servery. [15]

### 3.1.2 Funkcionalita integračního middleware

Integrační middleware je nadstavbou základního middleware. Nabízí rozšířenou funkcionalitu, která pomáhá sjednotit technické a aplikační odlišnosti v situacích, kdy spolu mají komunikovat či mají být vzájemně integrovány v zásadě nezávislé aplikační systémy [5]. Jelikož je integrační middleware rozšířením základního middleware, tak využívá funkcionalitu základního middleware. Integrační middleware zahrnuje následující oblasti [5]:

- adaptéry – adaptér je kombinace návrhového nástroje a běhového prostředí umožňující implementovat vhodné rozhraní pro komunikaci dvou prvků. Typicky zapouzdřuje původní rozhraní do nového, vyhovujícího rozhraní [5]. Adaptéry mají spíše charakter technických adaptérů, kam se např. řadí: adaptéry pro komunikaci do databáze, komunikační adaptéry pro MOM (message-oriented middleware) a platformy webových služeb, atd. Adaptéry se orientují na zapouzdření a vytvoření rozhraní mezi systémovými komponentami. [5]
- transformace – jedná se o prostředky, které zajišťují transformaci dat, tj. poskytují syntaktickou a sémantickou konverzi dat (dokumenty, zprávy), která jsou přenášena mezi aplikacemi. Syntaktická konverze představuje např. převod kódování, datových typů atd. Sémantická konverze vyžaduje již detailnější znalost aplikací a procesů a přináší obohacení či redukce původního obsahu. Kupříkladu pokud jedna aplikace předpokládá data ve tvaru jméno a příjmení a ta druhá aplikace ve tvaru příjmení a jméno. [5]
- inteligentní směrování – prostředky inteligentního směrování umožňují směrování zpráv mezi aplikacemi na základě obsahu a při použití definovaných pravidel směrovat tok k vhodnému příjemci. [5, 22]
- transparentní přístup k heterogenním datům – vytváří konzistentní pohled na různorodá data v různých úložištích bez znalosti jak k datům přistupovat a kde se data nacházejí. [5]
- řízení byznys procesů (BPM) – nabízí prostředky pro řízení, optimalizaci, automatizaci a transparentnost firemních i zákaznických procesů. Lze říci, že pokrývá aktivity spojené

s analýzou a definicí procesu, jeho vlastním prováděním, monitorováním a administrováním [5].

- řízení procesů na základě pravidel (BRE) – nabízí prostředky pro řízení podnikových prostředků takovým způsobem, že jsou schopny provést re-konfiguraci procesů (tj. změnit jejich chování) na základě definované množiny pravidel a politik [5].
- správu a dohled nad událostí (BEM) – tato oblast poskytuje prostředky, které jsou zaměřené na monitorování aktivit podnikových procesů. Jedná se o prostředky zachytávající a shromažďující informace o vzniklých událostech. V reálném světě jsou těmito představiteli agenti, kteří zachytávají a monitorují důležité aktivity, které následně mohou být vyhodnoceny. [5, 22]

### **3.1.3 Funkcionalita middleware pro aplikační integraci**

Tuto poslední kategorii tvoří čistě middlewarové prostředky, ale prostředky vnitřně integrující middleware s aplikační logikou zaměřenou na podnikové procesy [5]. Jedná se především o tyto kategorie:

- aplikační adaptéry - rozhraní aplikačních adaptérů zabalují celé aplikační moduly např. Oracle Financials, SAP Procurement, RosettaNet, S.W.I.F.T. atd [12]. Jde o zapouzdření původního (nevyhovujícího) rozhraní aplikace nebo aplikačního balíku.
- integrované balíky aplikací – tato oblast představuje specificky sestavované aplikační balíky, které zapouzdřují řešení nějakého konkrétního procesu, který vznikl na základě integrace více aplikací (PIP - Packard Integrating Process).
- business activity monitoring (BAM) - mechanismus BAM umožňuje snadno sledovat průběh obchodních procesů dle nasbíraných dat. Principiálně poskytuje v reálném čase přístup ke kritickým indikátorům výkonnosti byznys procesů, včetně možností v reálném čase ovlivňovat technologie tak, aby bylo možné zlepšit rychlost a efektivitu stěžejních obchodních operací [12]. Prostředky jsou často propojeny s business intelligence (BI) systémy.

## **3.2 Existující integrační nástroje**

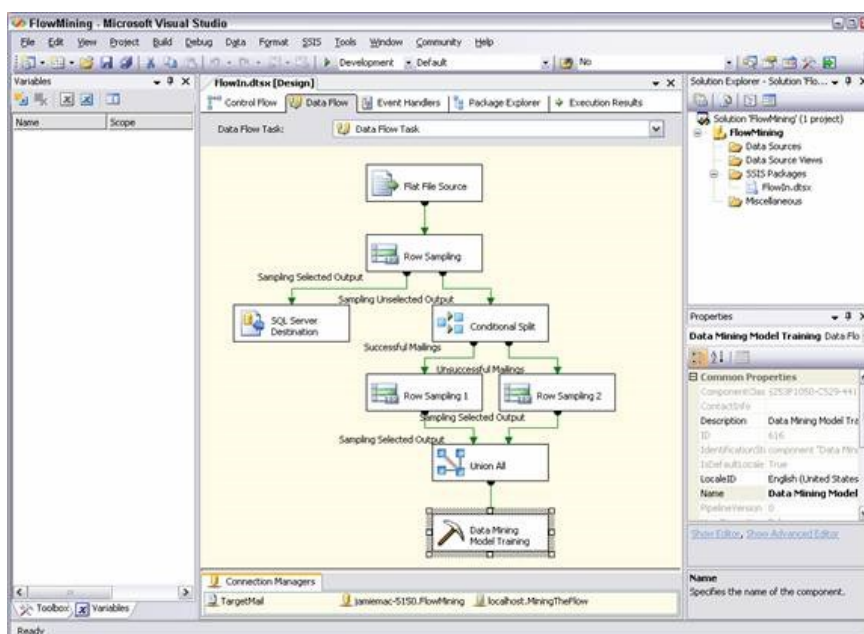
V následující kapitole jsou popsány jednotlivé technologie pro integraci dat, které by mohly být podle zadavatele použity pro vytvoření rozhraní mezi systémy ERP a WMS. Ačkoliv existuje velké množství dalších technologií zprostředkujících middleware, některé dostupné i ve volně šiřitelné verzi, pro přehlednost jsou zde zdokumentovány pouze technologie navržené zadavatelem. Níže zmíněné technologie se liší jak způsobem komunikace, tak složitostí struktury.

### 3.2.1 SQL Server Integration Server

Jedná se o nástroj pro podnikovou integraci a transformaci dat od společnosti Microsoft, která je vestavěna do prostředí Microsoft SQL Server Database. Tento nástroj může být využit pro velké množství rozdílných požadavků na integraci, mezi které patří například analýza dat, čištění dat, získávání dat z podnikových systémů, jejich následné zpracování a poskytování zpracovaných dat. Poslední tři zmíněné operace jsou označovány anglickou zkratkou ETL (extrude, transform, load). [13]

SSIS může získávat, transformovat a slučovat data z různých relačních databází i ze zdrojů, mezi které patří XML soubory a prosté databázové soubory, a následně zpracované informace nahrát do podnikového datového úložiště nebo jiného cílového systému. SSIS obsahuje sadu nástrojů pro vývoj a testování integračních programů nazývaných *SQL Server Data Tools* a komponentu serveru pro nasazení a spuštění programů pro integraci dat.

Jelikož je SSIS produktem společnosti Microsoft, je možné jednotlivé operace jako třeba transformace dat vytvářet ve vývojovém nástroji Visual Studio. V nástroji SSIS existují dva způsoby vývoje integračních služeb, které se navzájem nevyklučují, naopak se spíše používá kombinace obou postupů. Prvním způsobem vývoje je vizuální programování, kde člověk nemusí psát žádný programový kód v programovacím jazyce a vše jednoduše „nakliká“ v uživatelském prostředí Visual Studia. Druhým způsobem je naopak využití programovacího jazyka C# nebo Visual Basic pro programování funkcí například pro transformaci dat apod.



Obrázek 3 - Ukázka vizuálního programování SSIS balíčku  
<http://www.sqlserverdatamining.com/ssdm/Default.aspx?tabid=96&Id=338>

Nástroj SSIS poskytuje v prostředí Visual Studio mnoho dalších užitečných funkcí mezi které patří například odladění integrační aplikace pomocí spuštění v debug módu, sledování prostředků při komunikaci, odchyťování chyb a následné reagování na jejich výskyty.

### 3.2.2 BizTalk

BizTalk je dalším integračním nástrojem od společnosti Microsoft a má mnoho společného s integračním nástrojem SSIS. BizTalk má oproti SSIS větší množství konektorů na různé typy zdrojů dat a umožňuje přijímání zpráv z aplikací pomocí různých protokolů, zatímco SSIS je zaměřeno na získávání dat hlavně z databázových struktur.

Porovnání integračních nástrojů SSIS a BizTalk je shrnuto v tabulce 1

<b>BizTalk</b>	<b>SQL Server Integration Service</b>
Pro menší množství integračních dat.	Větší množství integračních dat založených na podobné technologii.
Plně založen na XML kvůli tomu pomalejší zpracování v případě velkého množství dat	Podporuje dávkové zpracování
Nabízí vyrovnanou integraci dat jak na vrstvě databázové, tak na vrstvě podnikové	Nabízí integraci na databázové vrstvě a podporuje podnikovou integraci pomocí SOAP (Simple Object Access Protokol).
Primárně zaměřen na XML konverzi.	Obsahuje konverzi mezi různými formáty souborů, například sešit Excel a databázová tabulka. Částečně podporuje i formát XML.
Obsahuje podmíněné a potvrzené schéma posílání zpráv.	Služba zprostředkovatele (broker) je používána pro asynchronní a konzistentní posílání zpráv. Je zde uplatňována fronta typu FIFO (First In First Out)
Obsahuje velké množství adaptérů jako například SAP pro podporu komunikace. Může a nemusí být závislý na připojení OLE DB.	Potřebuje přímou asociaci s databázovým OLE, případně s SQL databází v případě navázání komunikace s různými databázemi. Je plně závislé na OLE DB připojení.
Obsahuje vlastnost BAM (Business aktivity monitoring).	Funkčnost BAM musí být dodělána, což představuje komplexnější kód.
Pro transfer dat uvnitř aplikace je nutné vytvořit jejich reprezentaci, která je závislá na typu dat.	Transfer dat mezi databázovými tabulkami je prováděn na základě jednoduchých metod řídicích tok dat.
Musí být vytvořena reprezentace dat, která závisí na cílovém systému typu dat, která mají být transferována.	Tříděná i netříděná data mohou být importována pomocí obecných souborů.
Potřebuje licenci SQL Server k podpoře integrace dat.	Je součástí licence SQL Server.

Není postaveno pro ETL řešení, díky přenášení velkého množství dat.	Ideální nástroj pro ETL řešení.
Rychlý vývoj aplikací oproti SSIS.	Pomalý vývoj aplikací oproti BizTalk
Ideální pro integraci a migraci transakčních dat. Může pracovat v reálném čase.	Ideální pro integraci a migraci netransakčních dat.

Tabulka 1 - Porovnání technologií BizTalk a SSIS

Z tabulky je patrné, že použití nástroje BizTalk je vhodné hlavně při integraci dat mezi různými aplikacemi, které mohou být postaveny na rozdílných platformách. Pro účely vytvoření rozhraní mezi systémy ERP a WMS se hodí více i z hlediska možnosti zpracování dat v reálném čase. Důležitou součástí řešení je také možnost transferu transakčních dat a možnost synchronní komunikace. Nevýhodou použití tohoto nástroje je jeho vyšší cena a nutnost vlastnit licenci i na SQL Server.

### 3.2.3 SonicMQ

Jedná se o software od společnosti IBM, který je primárně vyvíjen pro použití s Java aplikacemi. Díky tomu umožňuje nasazení aplikace všude tam, kde běží Java virtuální stroj (JVM). Tento integrační middleware využívá konceptu výměny zpráv neboli sdílení zpráv (dat) mezi aplikacemi. Jedná se o JMS (Java Message Service) asynchronní výměnu zpráv. To znamená, že zasílající aplikace nečeká na přijetí a zpracování zprávy druhou stranou. Asynchronní zprávy tvoří autonomní jednotky, které v sobě obsahují vlastní data i veškeré další údaje potřebné pro komunikaci (hlavičku, stavy, přílohy, vlastnosti). Oproti tomu synchronní výměna zpráv znamená, že je odesílající aplikace blokována, dokud neobdrží potvrzení o přijetí od aplikace, která data přijímá. Synchronní komunikace je v konceptu zasílání zpráv použitelná, ale není obvyklá.

O zasílání a přijímání zpráv se stará klient zajišťující přenos zpráv. Architektura SonicMQ se tedy skládá z vlastního serveru a určitého počtu klientů. Klienti zasílají zprávy do SonicMQ serveru, který je rozesílá ostatním zaregistrovaným klientům. Klientem může být větší či menší aplikace nebo pouze komponenta, která používá rozhraní pro rozesílání zpráv. SonicMQ obsahuje velké množství komunikačních nástrojů včetně podpory obou modelů pro zasílání zpráv *point-to-point* a *publish-and-subscribe*. Na dynamicky se měnící počet připojených klientů, front dokáže SonicMQ server reagovat téměř lineárně. [14]

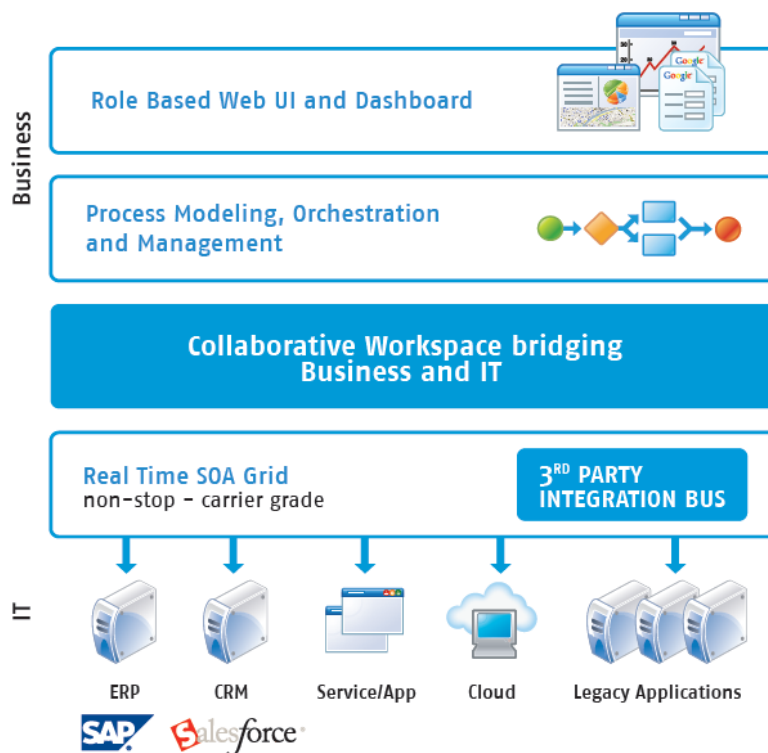
Vestavěný mechanismus primárního uložení zprávy a následné distribuce zprávy s interním protokolem pro potvrzování zpráv zaručuje doručení veškerých zpráv, které procházejí SonicMQ serverem. Zprávy mohou být ukládány do interní Java databáze či do dalších databází jako jsou databáze Progress, Oracle či MS SQL. Server také disponuje prostředky pro autentizaci, autorizaci (ACL) a zabezpečenou komunikaci. [14]

SonicMQ je zcela nezávislý na formátu přenášených zpráv (dokumentů), tudíž je vhodný pro přenos mezi aplikacemi postavenými na různých platformách. Zprávy se také dají seskupovat a zpracovávat v logických skupinách. Dokud není doručena i poslední zpráva, není možné jednotlivé zprávy zpracovávat. Tím je zaručena konzistence dat i v případě výpadku HW nebo dané aplikace.

Stejně jako je důležité mít přehled o zprávách doručených, neméně důležitý je pravý opak, tzn. mít přehled o zprávách, které se z různých důvodů nepodařilo doručit. SonicMQ obsahuje pro tyto účely zvláštní systémovou frontu mrtvých zpráv (*Dead Message Queue*), kterou lze prohlížet a vyhodnocovat. SonicMQ také umožňuje dynamicky přidávat nebo ubírat servery pro přenos zpráv, sdružovat je do klastrů a významně tak zvyšovat výkonnost a odolnost dané instalace. [14]

### 3.2.4 Cordys SOA Grid

Jak už je z názvu patrné, tento software je založen na konceptu SOA neboli architektuře orientované na služby. SOA je sada principů, metodologií a doporučení zajišťující funkčnost aplikace, která má být poskytnuta a konzumována jako veřejná sada služeb odpovídající potřebám zpracovatele. [17] Na obrázku 4 jsou znázorněny jednotlivé součásti aplikace Cordys SOA Grid. První dvě části slouží pro definování podnikových procesů pomocí uživatelského rozhraní, kde není nutné znát programovací jazyk. Tvorba procesů probíhá pomocí diagramů, jednotlivých adaptérů a dalších prostředků.



Obrázek 4 - Struktura Cordys SOA Grid



Cordys ESB je součástí produktu Cordys SOA Grid. Jedná se o rozšíření postaveném na architektuře webových služeb, které mají hlavní podíl při propojování rozdílných částí podnikových systémů. Cordys ESB nemá centrální hub a proto eliminuje možnost výskytu chyb v daném bodě a neoslabuje výkon systému v průchozím bodě.

Cordys ESB se skládá z následujících částí, které zaručují robustní řešení nezávislé na platformě:

- *Cordys Gateway* – slouží jako vstupní bod pro klienty přistupující ke službám přes internet, které jsou dostupné v podnikových aplikacích za firewallem.
- Rozhraní *Cordys Connectivity* - umožňuje organizacím připojit se k jejich stávajícím podnikovým aktivitám (jako jsou databáze, balíčkové aplikace, FTP servery a e-mailové servery) a jejich opětovné použití v nových kompozitních aplikacích.
- Rozhraní *Cordys Services* – umožňuje vývojářům vytvářet a udržovat životní cyklus služby na Cordys SOA Grid. Organizace mohou využívat širokou škálu komunikačních protokolů a protokolů zpráv k přístupu k nasazeným službám na SOA Grid.
- Rozhraní *Cordys Application Deployment* poskytuje robustní funkce pro velké množství typů nasazení aplikací obsahující podporu pro jednu instanci serveru, klastrové serverové modely a stejně tak i pro soukromé nebo veřejné hostující prostředí.
- *Cordys Single Sign On (SSO)* poskytuje plnou podporu pro jednotné přihlášení využívající SAML 1.1, které zajišťuje jednotný přístup k softwarovým prvkům skrze důvěrnou doménu s jedinečnou identitou.

Tento nástroj poskytuje velké množství možností integrace podnikových systémů, které poskytuje velké množství adaptérů, konektorů apod. Implementace v tomto prostředí ale vyžaduje hlubší studium dané problematiky integrace aplikací. Pro tvorbu rozhraní mezi dvěma aplikacemi ERP a WMS je toto řešení zbytečně složité a náročné na vývoj, tudíž i drahé a hodí se lépe pro integraci většího množství systémů dohromady, například ještě se systémy CRM, APS apod.

### **3.3 Volba technologie pro integraci ERP a WMS systému**

Po nastudování předchozích technologií poskytujících prostředky pro integraci dat, byla kvůli vysokým nákladům a složitosti struktury jednotlivých integračních nástrojů zvolena vlastní implementace integrační aplikace založené na principu webových služeb. V této části práce jsou webové služby představeny z obecného hlediska. Návrhem aplikace se následně detailněji zabývá kapitola 5.

Technologie webových služeb poskytuje řadu výhod. První výhodou je, že webové služby jsou nezávislé na použité platformě a programovacím jazyku, ve kterém byla vyvinuta samotná webová služba či přístupující aplikace. Jednotliví klienti webových služeb k nim mohou přistupovat standardním způsobem bez znalosti jejich platformy či programovacího jazyka. Další výhodou webových služeb je, že pracují se standardními webovými protokoly, mezi které patří XML, HTTP a TCP/IP. Webové služby jsou vystaveny na web, mohou komunikovat prostřednictvím SOAP protokolu a jsou popsány pomocí WSDL souboru.

### 3.3.1 Zprávy ve webových službách

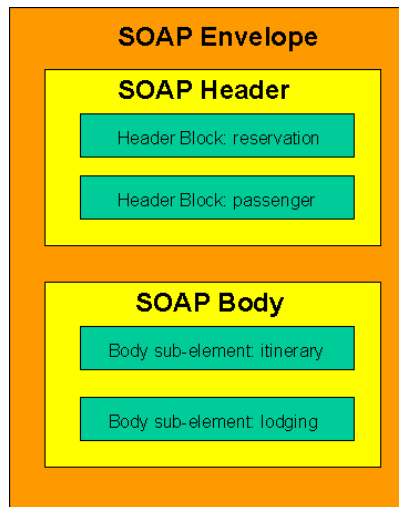
SOAP je anglickou zkratkou pojmu *Simple Object Access Protocol* neboli protokol pro snadný přístup k objektům. Jak již bylo zmíněno výše, SOAP je protokol pro posílání zpráv XML, který tvoří základ webových služeb.

Nejčastěji se SOAP používá jako náhrada vzdáleného volání procedur (RPC). To znamená, že využívá modelu požadavku a odpovědi. Vzor požadavku a odpovědi představuje následující scénář. Jedna aplikace pošle v XML zprávě požadavek druhé, dotazované aplikaci. Ta požadavek zpracuje a výsledek zašle jako druhou zprávu zpět původnímu iniciátorovi komunikace. Webové služby bývají vyvolány webovým serverem, který čeká na požadavky od klientů. Jakmile je přes HTTP přijat SOAP požadavek od klienta, server spustí webovou službu a předá jí požadavek. Následně je výsledek služby předán zpět klientovi jako odpověď.

První verze SOAP protokolu byla vytvořena při spolupráci firem DevelopMentor, Microsoft a UserLand v roce 1999. Jejich snahou bylo vytvořit protokol pro vzdálené volání procedur (RPC) založeném na XML.[6] O rok později byla zaslána definice SOAP protokolu do W3C konsorcia [7].

SOAP zprávu tvoří jednoduchý XML dokument, který obsahuje kořenový element <Envelope> (česky označován jako obálka). Uvnitř tohoto elementu jsou pak vnořeny další dva elementy. Prvním je nepovinný element <Header> (hlavička) a druhým je povinný element <Body> (tělo). Hlavička se používá pro přenos pomocných informací pro zpracování zprávy, mezi které patří například identifikace uživatele, autentizační informace (jméno, heslo) apod. Nejdůležitější částí SOAP zprávy je tělo, kde jsou obsaženy všechny informace identifikující volanou službu a předávané parametry, resp. návratové hodnoty služby. Tělo může obsahovat ještě jeden další element <Fault>, ve kterém jsou zaznamenány chyby při vykonávání dané funkce.

SOAP používá jmenné prostory pro identifikování jednotlivých částí XML zprávy. Obálka, hlavička a tělo zprávy patří do jmenného prostoru <http://schemas.xmlsoap.org/soap/envelope/>. [8]



Obrázek 5 - Struktura SOAP zprávy [8]

### 3.3.2 Dokument popisující webovou službu

WSDL je zkratkou pro anglický název *Web Services Description Language* a už z anglického názvu vyplývá, že se jedná o dokument popisující webovou službu. Tento dokument je popsán pomocí jazyka XML a specifikuje umístění webové služby a operace (metody), které tato služba vystavuje. Operace jsou popisovány na abstraktní úrovni a teprve poté jsou svázány s konkrétním síťovým protokolem a datovým formátem. To umožňuje snadné vytvoření popisu rozhraní, které nabízí jednu službu několika způsoby. V praxi WSDL popisy nejčastěji popisují služby, které si posílají zprávy pomocí formátu SOAP a protokolu HTTP.

WSDL vzniklo jako společná iniciativa firem Microsoft a IBM, které si uvědomily potřebu sjednocení jazyka používaného pro popis rozhraní webových služeb. Jazyk WSDL spadá pod konsorcium W3C, kam byl přidán v roce 2007 [9.]

WSDL se skládá z následujících elementů, které tvoří základní části každého WSDL popisu [9]:

**Types** - Obsahuje definici datových struktur používaných ve zprávách. K definici je možné použít libovolný typový systém, ale nejčastěji se používají XML schémata. Nástroje pro webové služby se starají o mapování datových typů podle XML schémat na nativní datové typy použitého jazyka.

**Message** - Definuje formát předávaných zpráv pomocí dříve definovaných datových typů. Zprávy fungují jako vstupní anebo výstupní struktury pro operace. Každá zpráva může být složena z několika logických částí s vlastním datovým typem. Při použití SOAP zpráv pro RPC odpovídá jedna část zprávy jednomu parametru vzdálené metody.

**Operation** - Abstraktní definice operací, které jsou službou podporovány. U operace se definuje jaké má vstupy neboli parametry a jaké má výstupy neboli návratovou hodnotu. Vstup

a výstup je popsán již existující zprávou (`message`). V SOAP RPC modelu odpovídá operace metodě.

`portType` - Sdružuje dohromady několik operací.

`Binding` - Slouží pro navázání určitého typu portu (`portType`) na konkrétní protokol a formát přenosu zpráv.

`Port` - Jedná se o jeden koncový bod služby definovaný jako kombinace síťové adresy a dříve definované vazby (`binding`).

`Service` - Sdružuje několik koncových bodů (portů) do jedné služby.

## 4 Analýza integrační aplikace

Tato kapitola je zaměřena na definici pojmů používaných při skladových operacích a popisuje skladové procesy v systémech ERP a WMS. V další části jsou analyzovány jednotlivé případy užití a nastíněna vzájemná komunikace mezi systémy. Tím jsou specifikovány jednotlivé požadavky na integraci obou systémů. Analýza skladových operací je důležitou součástí pro správné určení rozhraní a odesílaných dat mezi systémy.

### 4.1 Definice pojmů užívaných při skladových operacích

Aby bylo možné přesně specifikovat požadavky na univerzální interface mezi systémy ERP a WMS, je nutné přesně specifikovat jednotlivé pojmy související se skladovými operacemi. V každém systému mohou být použité různé názvy pro stejné operace, hlavně u specifických rozšíření jako je synchronizace skladu apod. Proto jsou v této kapitole definovány používané definice skladových operací.

#### **Příjmová zóna**

Jedná se o místo na skladě sloužící pro vyložení zboží z nákladního automobilu. Zde je zboží kontrolováno a nachystáno k přesunutí na sklad.

#### **Příjem zboží**

Příjem zboží je proces, kdy je zboží přijato od dodavatele zboží do firmy. Fyzicky se jedná o vyložení zboží od přepravce do příjmové zóny skladu na základě nákupní objednávky vytvořené v systému ERP. Může se jednat o příjem od přímého dodavatele, z pobočky, vratky od zákazníka apod. Při příjmu se zboží stává majetkem společnosti, ale ještě s ním není možné dále operovat, jelikož se ještě nestává disponibilním.

#### **Zaskladnění**

Jedná se o proces, při kterém, je zboží fyzicky i účetně přemístěno z příjmové zóny skladu do konkrétního umístění na skladu neboli do skladových či jiných speciálních přihrádek. Po zaskladnění se již zboží stává manipulovatelným.

#### **Přeložení zásilky**

Jedna se o proces, někdy označovaný jako crossdock, kdy je zboží přesunuto z příjmové zóny skladu přímo do dodací zóny skladu bez toho, aby se zboží uložilo do vyskladňovací zóny skladu. To znamená, že se přeskočí proces zaskladnění a vyskladnění do a ze skladové přihrádky. Přeložení zboží může být použito například u objednávky z lokálního (podřízeného) skladu nebo přímo zákazníkovi, která je ale nejdříve přijata centrálním skladem a poté přeposlána na daný lokální sklad.

## **Dodací zóna**

Jedná se o místo na skladě, kde je zboží nachystáno k naložení na nákladní automobil. Před naložením musí být zkompletováno a zkontrolováno.

## **Výdej**

Výdej je proces, při kterém je přesunuto zboží z firmy zákazníkovi apod. Jedná se o přesun zboží z dodací zóny na nákladní automobil, který zboží doručí danému zákazníkovi nebo na jinou lokaci. Výdej zboží je prováděn na základě zákaznické objednávky či objednávky transferu z jiné lokace v systému ERP.

## **Vyskladnění**

Proces, při kterém je zboží přemístěno z vyskladňovací zóny do dodací zóny skladu. Zboží je stále ve skladu firmy, ale je již nachystáno na naložení na nákladní automobil.

## **Noční synchronizace**

Při noční synchronizaci se porovnává účetní množství všech zboží na skladě obou systémů WMS i ERP. To je důležité pro zachování konzistence dat a k zachování co nejpřesnější informace o dostupnosti zboží. Název noční synchronizace je dán tím, že se jedná o porovnávání skladů v předem dohodnutém a pravidelném intervalu, který je nastaven v době nečinnosti skladů, a to bývá nejčastěji právě v noci.

## **Denní synchronizace**

Denní synchronizací se rozumí fyzická kontrola množství určitého zboží na skladu. Uživatel musí projít všechny přihrádky s daným zbožím a zaznamenat jeho množství. Často se při denní synchronizaci prohlížejí i sousední přihrádky, aby se zjistilo, zda nebylo zboží odloženo do špatné přihrádky. Denní synchronizace je prováděna ihned po zjištění nesprávného množství zboží ve skladu (na přihrádce apod.).

## **Diference**

Diference vznikají, pokud fyzické množství zboží neodpovídá účetnímu množství. Při diferenci je zboží přesouváno například na fiktivní přihrádku systému ERP, která je dále v práci označována jako adjustační. Příslušný pohyb zboží je v takovém případě dobré doplnit o důvod přesunu (například v aplikaci Dynamics NAV je to označováno jako kód příčiny). Obsah adjustační přihrádky představuje interní blokaci zboží, zabraňující příslibení zboží zákazníkovi a manipulaci zboží ve skladu. Nejedná se o rezervaci, ale navenek je jako u rezervace odečteno množství od disponibilního množství.

### **Adjustační zóna**

Adjustační zóna je virtuální zóna sloužící k vyrovnávání diferencí zboží na skladě. Jedná se o vyrovnání diferencí, tudíž při plusové diferenci se zboží z adjustační zóny odebírá, zatímco u minusové difference se zboží na adjustační zónu vkládá. Tato adjustační zóna je nulována jednou za rok při velké inventuře skladu nebo při vložení zboží do adjustační přihrádky na jiné lokaci.

### **Konsignační zóna**

Jedná se o zónu, kde je uložené zboží stále majetkem dodavatele. Sklad s ním může operovat jako s disponibilním zbožím i přes to, že není účetně jeho vlastníkem. Záleží na domluvě mezi partnery, kdy se zboží stane majetkem skladu. Například při přesunutí z konsignační do skladové nebo dodací zóny se stává zboží majetkem společnosti.

### **Dostupné množství zboží**

Dostupným množstvím zboží se rozumí veškeré množství zboží, které se aktuálně nachází v zásobách společnosti – patří sem zboží na příjmu, zboží v zásobních zónách, zboží ve vyskladňovací zóně, zboží v dodací zóně, v přepravě a zboží na všech pobočkách. Toto množství je zobrazováno jako "Zásoby". Dostupné množství je směrodatný ukazatel pro potřeby nákupu, který toto množství doplňuje. Dostupné množství je při časovém vývoji navýšeno nákupem, ponižováno prodejem a přesunem.

### **Disponibilní množství zboží**

Disponibilní množství je množství volných zásob, které mohou být nabídnuty zákazníkům k rychlému odběru. Patří sem nerezervované zboží ve vyskladňovací zóně nebo v přepravě a nerezervované zboží na všech pobočkách. Toto množství je označováno jako disponibilní. Disponibilní množství je směrodatný ukazatel pro potřeby prodeje při stanovení termínů dodání a příslibování rezervací zákazníkům. Disponibilní množství je navyšováno při zaskladnění po příjmu zboží a skladovými přesuny a je ponižováno rezervací nejen pro zákazníka.

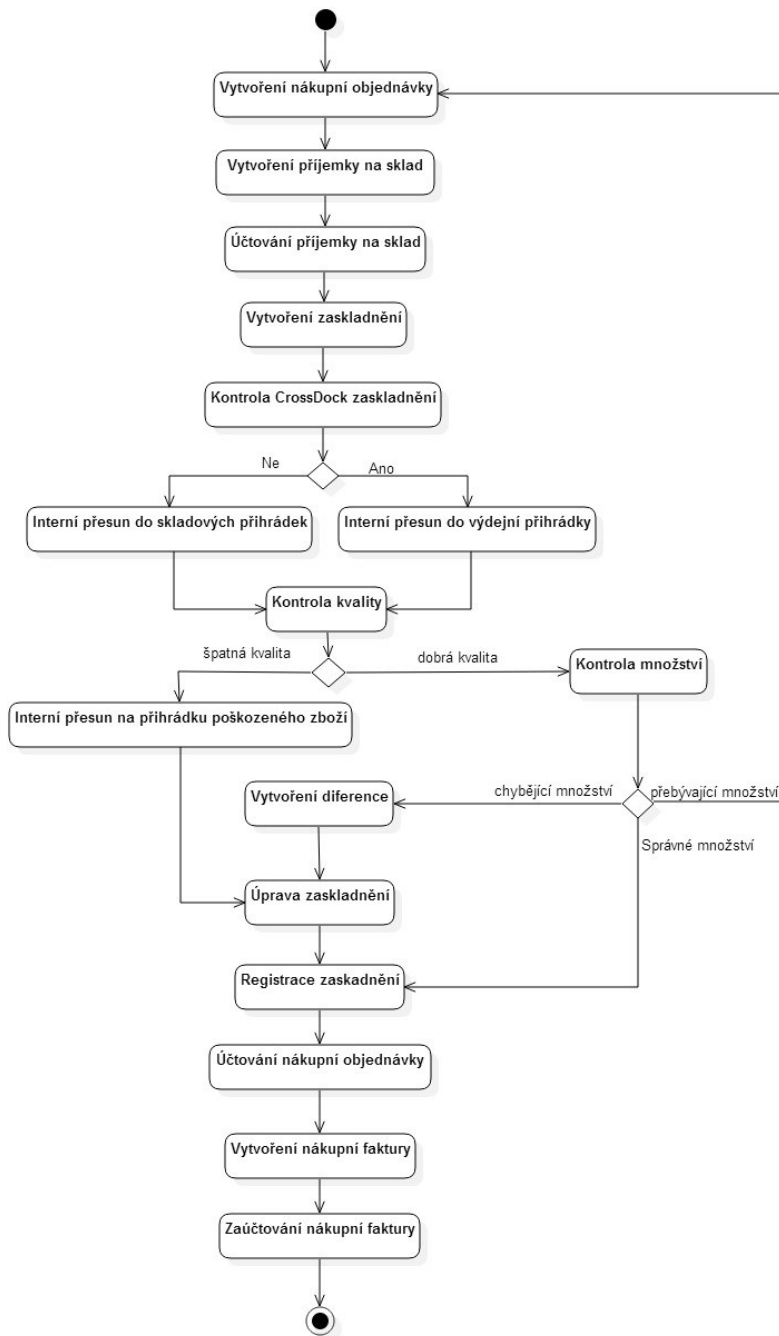
## **4.2 Jednotlivé případy užití a komunikace mezi systémy**

V následující kapitole jsou definovány jednotlivé případy užití při základních skladových operacích, jako jsou příjem, výdej a inventura. Příjem a výdej zboží je nejčastější příčinou pro výměnu informací mezi jednotlivými systémy. Při těchto skladových operacích musí zůstat data v obou systémech konzistentní.

Nejčastější formou dat v systémech ERP jsou záznamy označované jako dokumenty nebo doklad. Každý dokument se skládá z hlavičky a řádků. Hlavička obsahuje společné informace pro všechny řádky dokumentu. Každý řádek dokumentu potom obsahuje detailní informace, které se vztahují pouze k jednomu určitému zboží.

### 4.2.1 Příjem zboží

V tabulce je popsána případová studie příjmu zboží na sklad. Příjem zboží na sklad zahrnuje mnoho podprocesů, které jsou popsány níže v textu pod tabulkou, a které s ním přímo souvisejí. Průběh příjmu zboží z venku do firmy je graficky znázorněn pomocí UML aktivitního diagramu na obrázku 6.



Obrázek 6 - Aktivitní diagram pro příjem zboží



V následující tabulce případu užití (Tabulka 2) se popisují přístupy jednotlivých systémů při komunikaci mezi systémy a následné zpracování požadavků.

Název případu užití	Příjem zboží	
Identifikace případu užití	UC1	
Cíl případu užití	Registrace dokladu zaskladnění v systému ERP	
Primární aktér(ři)	WMS, ERP	
Vstupní podmínky	V systému ERP je zaúčtována příjemka na sklad	
Výstupní podmínky	Doklad o zaskladnění je zpracován systémem WMS a výsledky jsou poslány do systému ERP k zaregistrování.	
Základní scénář		
Krok	Role	Akce
1	ERP	Vytvoření příjemky na sklad.
2	ERP	Zaúčtování příjemky na sklad.
3	ERP	Vytvoření dokladu zaskladnění.
4	ERP	Odeslání dokladu zaskladnění do WMS.
5	WMS	Přijetí dokladu zaskladnění.
6	WMS	Kontrola množství přijímaného zboží.
7	WMS	Kontrola kvality.
8	WMS	Záznamy z kontrol jsou zapsány do dokumentu zaskladnění.
9	WMS	Odeslání výsledků zaskladnění do ERP.
11	ERP	Příjem odeslaného dokladu zaskladnění z WMS.
12	ERP	Potvrzení přijetí dokladu zaskladnění.
13	ERP	Registrace dokladu zaskladnění.
Četnost užití:	Každou minutu	
Priorita:	Vysoká	

Tabulka 2 - Příklad užití pro příjem zboží

### ***Vytvoření příjemky na sklad***

Příjemka na sklad je vytvářena ERP systémem na základě informací z nákupní objednávky, objednávky transferu, vratky od zákazníka. Tento dokument obsahuje mimo jiné i informace o původním dokumentu, to znamená jeho typu (nákupní objednávka, transfer z jiné pobočky apod.). Na příjemku na sklad jsou zaznamenávány důležité informace o přijetí zásilky, např. čas příjezdu automobilu se zbožím, druh příjmu (normální zaskladnění nebo přeložení zboží), atd. V řádcích jsou potom obsaženy informace týkající se jednotlivého zboží, které má být zaskladněno.

### ***Zaúčtování příjemky na sklad***

Při zaúčtování příjemky na sklad vznikne pohyb zboží dovnitř firmy. Následně je v systému ERP vytvořen dokument zaskladnění, obsahující základní informace o zboží a jeho následných přesunech v rámci skladu. V řádcích dokumentu zaskladnění bývají informace týkající se množství zboží, které má být zaskladněno, pravděpodobné místo odkud se bude zboží přesouvat a místo uložení zboží (skladová zóna, zóna pro přeložení zboží). Systém ERP následně pošle informaci o vytvoření dokladu zaskladnění systému WMS. Na základě tohoto dokumentu začne systém WMS vykonávat potřebné operace k přemístění zboží z příjmové přihrádky do konkrétních umístění na skladu.

### ***Proces zaskladnění***

Po přijetí dokladu zaskladnění systémem WMS začíná fáze přesunu zboží z příjmací přihrádky skladu na skladové přihrádky. Při provádění zaskladnění je vykonáno mnoho dalších dílčích procesů, jako jsou kontrola kvality a množství zboží. Jednotlivé výsledky jsou zaznamenány do dokladu zaskladnění, kde se zaznamenává pouze fyzicky zpracované zboží, a pokud se zpracované množství liší od požadovaného, zapíše se důvod rozdílu do dokladu. Po zaskladnění zboží a zápisu odpovídajících informací do dokumentu zaskladnění proběhne odeslání výsledků zpět do systému ERP.

### ***Kontrola množství zboží***

Při příjmu zboží je provedena kontrola množství zboží systémem WMS, zda fyzické množství zboží odpovídá množství, které je uvedeno v dokumentu zaskladnění. Pokud se tyto dvě položky neshodují, systém WMS zapíše fyzicky zaskladněné množství zboží na dokument zaskladnění a zapíše důvod přijetí většího či menšího množství zboží. Po odeslání do systému ERP jsou na základě těchto informací vytvořeny záznamy o diferencí.

### ***Vytvoření diferencí***

V případě, že je skladem přijato menší množství od dodavatele, než je definováno v dokladu příjemky, je systémem ERP vytvořena mínusová diference s rozdílným množstvím zboží. Pokud se jedná o případ, kdy je přijato větší množství zboží, než je uvedeno v dokladu příjemky, pak je vytvořena plusová diference.

### ***Zpracování diferencí***

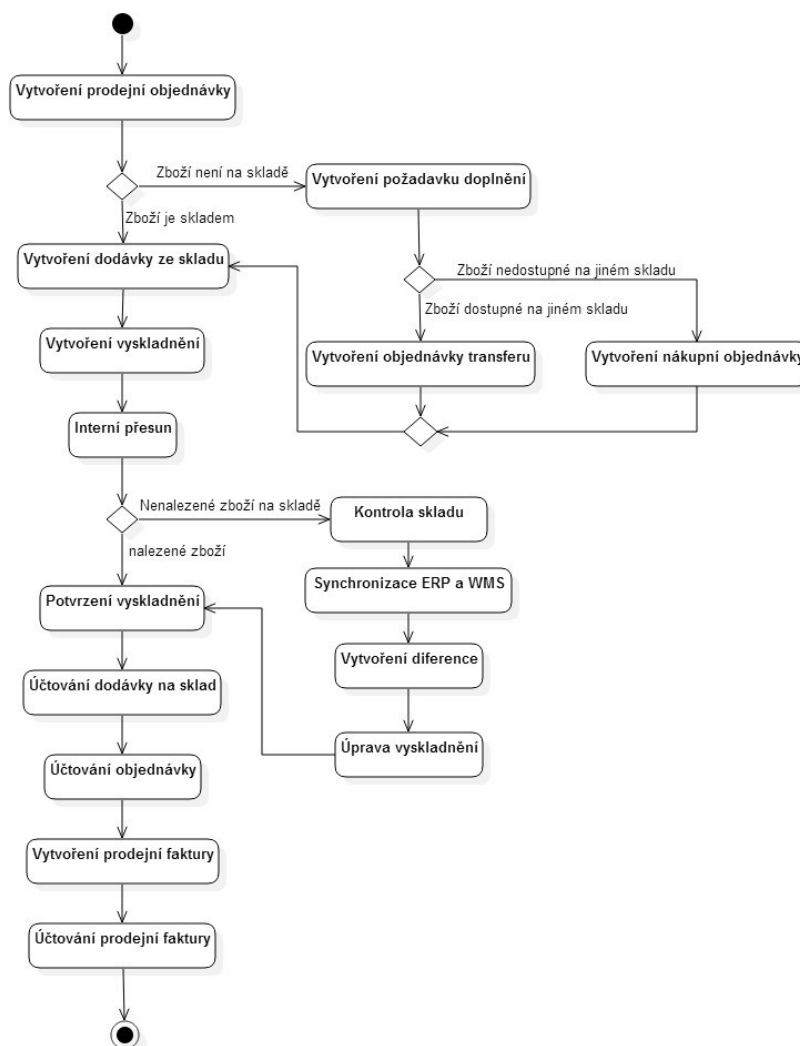
Zpracování diferencí probíhá v systému ERP. Při zpracování mínusové diference je chybějící množství zboží přidáno do adjustační přihrádky. V opačném případě, při plusové diferencí, je zboží vyjmuto z adjustační přihrádky a přesunuto na skladovou přihrádku. Pokud se zpracovává plusová diference z příjmu, neboli nefakturovaná, pak je v systému ERP vytvořena nová nákupní objednávka, která obsahuje množství zboží, které bylo doručeno navíc, a celý proces pokračuje standardním způsobem. To znamená vytvoření skladové příjemky, dokladu zaskladnění, přeposlání do systému WMS atd. Po zpracování nákupní objednávky je zboží zafakturováno a diference je označena za vyřešenou.

### Registrace dokladu zaskladnění

Registrace dokumentu zaskladnění probíhá v systému ERP na základě přijatých záznamů o zaskladnění z WMS. Označuje, že se již zboží nachází v přihrádkách skladu a tudíž se stává disponibilní pro další operace v rámci skladu (například pro vyskladnění či jiný skladový přesun).

### 4.2.2 Výdej zboží

Výdej zboží je dalším důležitým případem užití, který se skládá z mnoha dalších podprocesů v obou systémech. Při výdeji zboží existuje také mnoho faktorů ovlivňujících jeho průběh. Jednotlivé faktory jsou popsány pod tabulkou. Průběh výdeje zboží zachycuje aktivitní diagram na obrázku 7.



Obrázek 7 - Aktivitní diagram pro výdej zboží

V následující tabulce případů užití (Tabulka 3) lze vidět komunikaci mezi jednotlivými systémy WMS a ERP při provádění výdeje ze skladu.

Název případu užití	Výdej ze skladu	
Identifikace případu užití	UC2	
Cíl případu užití	Zaúčtování dodávky ze skladu	
Primární aktér(ři)	WMS, ERP	
Vstupní podmínky	V ERP je vytvořena dodávka ze skladu	
Výstupní podmínky	Dodávka ze skladu je zaúčtována.	
Základní scénář		
Krok	Role	Akce
	1 ERP	Vytvoření dodávky ze skladu.
	2 ERP	Vytvoření dokladu vyskladnění.
	3 ERP	Odeslání dokladu vyskladnění do WMS.
	4 WMS	Přijetí dokladu vyskladnění.
	5 WMS	Kontrola a přesun zboží ze skladu na dodací přihrádku.
	6 WMS	Potvrzení naložení zboží na nákladní automobil.
	8 WMS	Odeslání výsledků dokladu vyskladnění do ERP.
	9 ERP	Příjem výsledků vyskladnění z WMS.
	10 ERP	Úprava dodávky ze skladu podle informací z dokladu vyskladnění.
	11 ERP	Zaúčtování dodávky ze skladu.
Alternativní scénář:	Nenalezení zboží	
Krok	Role	Akce
	5a WMS	Nenalezení dostatečného počtu zboží na skladových přihrádkách.
	6a WMS	Provedení denní synchronizace skladu.
	7a WMS	Vytvoření záznamu difference na základě denní synchronizace
	8a WMS	Odeslání záznamu difference do systému ERP.
	9a ERP	Zpracování difference.
	10a WMS	Úprava dokumentu vyskladnění.
Četnost užití:	Každou minutu a častěji	
Priorita:	Vysoká	

Tabulka 3 - Příklad užití pro výdej zboží

### ***Vytvoření dodávky ze skladu***

Systém ERP vytváří dokument dodávky ze skladu podle informací z prodejní objednávky, objednávky transferu a podobně. Z dokumentu dodávky ze skladu se následně vytváří a odesílá

doklad vyskladnění. Takový dokument může obsahovat více položek z různých objednávek, které mají být vyskladněny.

### ***Vytvoření vyskladnění***

Dokument vyskladnění je generován systémem ERP podle informací z dokumentu dodávky ze skladu. V dokumentu vyskladnění mohou být uvedeny mimo jiné informace o typu vyskladnění (například minimax, zákaznická apod.) a další upřesňující informace důležité pro systém WMS k provedení vyskladnění.

### ***Odeslání vyskladnění do WMS***

Tento proces je inicializován uživatelem na straně ERP. O vytvoření dokumentu vyskladnění v systému ERP musí být informován systém WMS aby mohl začít se zpracováním položek uvedených na daném dokladu. Proto je dokument vyskladnění odeslán do systému WMS, který do něj poté zapisuje jednotlivé kroky prováděné při zpracování vyskladnění.

### ***Odeslání minimax do WMS***

Minimax je transfer mezi pobočkami pro doplnění zboží na lokálním skladu, aby disponibilní množství odpovídalo rozmezí minimálního a maximálního množství, které je definováno pro konkrétní zboží. Tento proces je inicializován na straně ERP. Před odesláním vyskladnění typu minimax bývá zpravidla vytvořena jeho rezervace, která je pevně spojena s dokumentem dodávky. Z dokumentu dodávky na sklad je vytvořen dokument vyskladnění, který obsahuje kromě základních informací také číslo rezervace, se kterou je tento dokument propojen. Dokument vyskladnění je následně poslán do systému WMS, který podle čísla rezervace zjistí, zda se jedná o minimax a přiřadí prioritu vyskladnění.

### ***Zaúčtování dokladu vyskladnění***

Zaúčtování dokladu vyskladnění je provedeno v systému WMS. Po zaúčtování dokumentu vyskladnění jsou informace odeslány do systému ERP a označeny jako zpracované. Jakmile WMS odpoví na všechny řádky dokladu vyskladnění, může systém ERP zaúčtovat dodávku na sklad.

### ***Uzavření dokumentu o naložení zboží na nákladní automobil***

Uzavření dokumentu o naložení nákladního automobilu probíhá v systému WMS na základě dokladu vyskladnění. Tento dokument označuje, kdy byl nákladní automobil naložen, které zboží bylo naloženo, a které zboží naopak naloženo nebylo. Informace o naložení zboží na nákladní automobil jsou odeslány do systému ERP. Systém ERP rozhodne, jak se zachovat, pokud není naloženo všechno zboží z dokladu vyskladnění. První možností je odeslání zboží s příští dodávkou ze skladu, druhou možností je zrušení dodávky ze skladu a následné vyskladnění zboží z výdejové přihrádky do skladu.

### ***Nenalezení zboží na skladě při vyskladnění***

Pokud skladový operátor nenalezne zboží, které je uvedené na dokladu vyskladnění na žádné přihrádce skladu, je v systému WMS spuštěna denní synchronizace. V rámci denní synchronizace je informace o skutečném stavu zboží odeslána ze systému WMS do systému ERP. Systém ERP vyplní informace o vyskladněném zboží a zboží k dodání do odpovídajícího řádku dodávky ze skladu a následně jej označí jako uzavřený.

### ***Zaúčtování dodávky ze skladu***

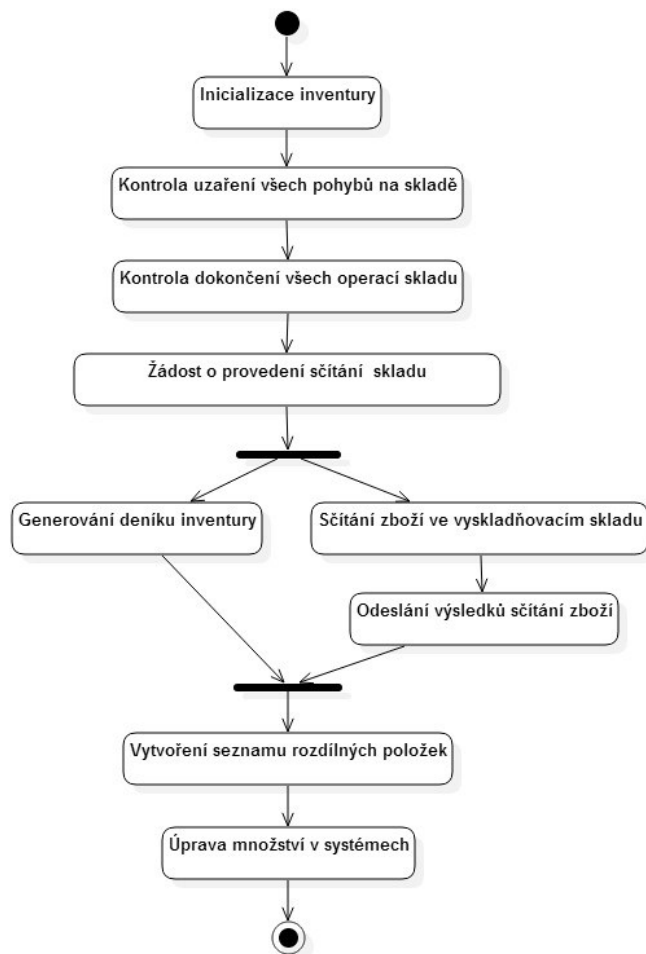
Zaúčtování dodávky ze skladu je provedeno v systému ERP na základě informací obdržených z účtovaného dokladu vyskladnění, který byl přijat ze systému WMS. Mohou být zaúčtovány pouze řádky dodávky na sklad, které jsou označeny jako naložené a odeslané. Dokument dodávky ze skladu je po zaúčtování v systému ERP smazán v případě plného vyskladněného dokladu.

### ***Přesun poškozeného zboží***

Mohou se vyskytovat dva případy nalezení poškozeného zboží. Prvním je nalezení zboží při zaskladnění. V takovém případě musí být systém ERP informován o nalezení poškozeného zboží a při jaké operaci k tomuto nálezu došlo. Druhým případem je nalezení poškozeného zboží při manipulaci na skladě. V tomto případě je nutné přesunout zboží do místa určeného pro poškozené zboží.

## **4.2.3 Inventura**

Inventura se v řízených skladech může rozdělovat na hlavní inventuru a částečnou inventuru. Hlavní inventura bývá prováděna nejčastěji jednou ročně. Částečná inventura se může zaměřovat pouze na kontrolu množství zboží určité kategorie nebo zboží od jednoho či více dodavatelů. Průběh obou inventur je podobný. Aktivitním diagramem je popsán proces celkové inventury skladu. Částečná inventura je o něco jednodušší a není nutný její podrobný popis.



Obrázek 8 - Aktivitní diagram pro celkovou inventuru skladu

Tabulka případu užití pro celkovou inventuru skladu:

Název případu užití	Inventura skladu	
Identifikace případu užití	UC3	
Cíl případu užití	Provedení inventury zboží skladu	
Primární aktér(ři)	WMS, ERP	
Vstupní podmínky	Při velké inventuře vyskladňovacích přihrádek musí být blokovány všechny pohyby na skladu. V ERP nesmí existovat nedoúčtované dodávky, přesuny apod.	
Výstupní podmínky	Sjednocení účetního stavu skladu v ERP s fyzickým stavem skladu ve WMS.	
Základní scénář		
Krok	Role	Akce
1	ERP	Inicializace inventury skladu

2	ERP	Blokace všech pohybů na skladě.
3	ERP	Žádost o provedení inventury vyskladňovacího skladu
4	WMS	Kontrola dokončenosti všech pohybů na skladu.
5	WMS	Sčítání zboží vyskladňovacích přihrádek skladu.
6	WMS	Odeslání výsledku inventury s celkovým počtem jednotlivého zboží
7	ERP	Porovnání výsledků inventury z WMS s účetním stavem ERP
8	ERP	Změna rozdílných položek v deníku zboží.
Četnost užití:		Jednou za rok
Priorita:		Vysoká

Tabulka 4 - Příklad užití pro celkovou inventuru skladu

Hlavní inventura se provádí nejčastěji na konci účetního roku. Skladové operace mohou být přerušeny i na několik dní, kdy nemohou být vyřizovány žádné objednávky a ani jiné pohyby na skladu. Naproti tomu částečná inventura skladu nevyžaduje blokování všech pohybů na skladě. Průběh komunikace je zachycen v tabulce 5.

Název případu užití	Inventura skladu	
Identifikace případu užití	UC3	
Cíl případu užití	Provedení částečné inventury zboží skladu	
Primární aktér(ři)	WMS, ERP	
Vstupní podmínky	Při sčítání nesmí být prováděny a ani otevřeny žádné pohyby na skladu.	
Výstupní podmínky	Sjednocení účetního stavu skladu v ERP s fyzickým stavem skladu ve WMS.	
Základní scénář		
Krok	Role	Akce
1	ERP	Inicializace částečné inventury (specifikace dodavatele, čísla zboží apod.)
2	WMS	Fyzický přepoččet zboží na skladě.
3	WMS	Odeslání výsledků do ERP
4	ERP	Příjem a porovnání výsledků
5	ERP	Úprava dostupnosti zboží
Četnost užití:	Několikrát do měsíce	
Priorita:	Vysoká	

Tabulka 5 - Příklad užití pro částečnou inventuru skladu



V tabulkách je uveden obecný postup inventury skladu. Níže jsou jednotlivé procesy popsány podrobněji.

#### ***Inicializace inventury skladu***

Systém ERP pošle žádost o inicializaci inventury skladu systému WMS. Informace o inicializaci inventury skladu musí obsahovat údaje jako je například datum, kdy má být inventura zahájena a druh o jakou inventuru se jedná. V případě částečné inventury jsou systémem ERP specifikovány detailní informace o kategorii zboží, čísla dodavatele apod. V případě celkové inventury skladu nejsou specifikovány žádné podrobnější informace. Následně jsou informace odeslány do systému WMS.

#### ***Výpis účetního stavu množství zboží v systému ERP***

Systém ERP si po zahájení inventury generuje seznam zboží s účetním disponibilním množstvím zboží skladu. Tento vygenerovaný seznam poté porovnává s obdrženými výsledky fyzického sčítání skladu v systému WMS.

#### ***Kontrola dokončených skladových operací***

Před zahájením samotného sčítání zboží musí systém WMS zkontrolovat, zda neexistují nějaké otevřené pohyby na skladě. To znamená, že v systému nesmí být žádné nezaúčtované doklady zaskladnění či vyskladnění.

#### ***Sčítání zboží na skladu***

Po kontrole neexistence otevřených skladových dokumentů může začít samotné sčítání zboží na skladě. Jakmile je zboží sečteno ve všech přihrádkách skladu, zapíše se celkový počet kusů zboží do dokladu ke sčítanému zboží. Tyto hodnoty jsou uchovávány v systému WMS.

#### ***Odeslání výsledků do systému ERP***

Pokud je již sčítání zboží kompletní, mohou být výsledky poslány do systému ERP. Systém po obdržení výsledků fyzického sčítání začne s porovnáváním účetního a fyzického množství zboží na skladě. Při nalezení nesrovnalostí srovná ERP systém účetní množství zboží podle skutečného množství odeslaného ze systému WMS.

### **4.2.4 Další skladové operace**

Následující podkapitola popisuje jednotlivé skladové operace, které jsou společné jak pro systémy WMS, tak pro systémy ERP. Následující procesy se mohou vyskytovat buď samostatně, třeba zařazení nového zboží do nabídky, nebo v rámci nějakého nadřazeného procesu.

#### ***Noční synchronizace***

Při noční synchronizaci musí být všechny pohyby zboží propsány v obou systémech WMS a ERP a ukončeny. Každý den v určenou hodinu se systém ERP i WMS zablokuje proti přesunům na skladu a systém ERP vygeneruje seznam skladovaného zboží. Tento seznam je poté porovnán

se seznamem zboží obdrženém ze systému WMS. Pokud se nějaké položky neshodují, vytvoří záznam diference. Následující den probíhá denní synchronizace v systému WMS podle záznamů diferencí a jejich následné zpracování.

### ***Denní synchronizace***

Denní synchronizace může být zahájena z důvodu nalezení diferencí při noční synchronizaci nebo při nenalezení určitého zboží na skladu při vyskladnění. Kontrolor skladu provede fyzické sčítání zboží, uvedeného v diferencích a ověří fyzickou dostupnost na skladě. Po sečtení množství daného zboží na všech přihrádkách se odešle celkový počet zboží uloženého na skladě systému ERP. V systému ERP je následně diference zpracována příslušným uživatelem. Ten může takovou diferenci buď přijmout a přesunout položku zboží do nebo z adjustační zóny, nebo může diferenci odmítnout, například za účelem provedení další fyzické kontroly zboží.

### ***Přidání nebo změna zboží***

Pokud je v systému ERP evidované nové zboží, pak musí být nové informace odeslány do systému WMS. Podobné je to i u úprav informací o zboží, které musí být propsány do systému WMS. Například se jedná o změnu skupiny, kategorie, čárový kód, měrné jednotky zboží a podobné.

### ***Přidání nebo změny údajů o zákazníkovi nebo o dodavateli***

Při změně údajů o zákazníkovi nebo dodavateli v systému ERP nebo při vytvoření nového zákazníka či dodavatele v systému ERP musí být systém WMS neprodleně informován a patřičná data musí být propsána.

## 5 Implementace

Následující kapitola se zabývá samotným návrhem integrační aplikace a následné implementace. Popisuje datovou strukturu aplikace, použité aplikační rámce, návrhové vzory a zvolenou technologii pro přenos dat mezi aplikacemi.

Webová služba bude hostována na softwarovém webovém serveru (IIS) operačního systému Windows od společnosti Microsoft. Tato webová služba bude vystavovat dva koncové body pro klienty, které budou dostupné pomocí protokolu HTTP.

První koncový bod bude vyhrazený pro klienta aplikace ERP, zatímco druhý koncový bod bude využíván aplikací WMS. Každý z koncových bodů má přesně definované dostupné funkce webové služby, které odpovídají danému koncovému bodu podle typu klienta. Přístup každého klienta k webové službě bude zajištěn pomocí autentizace v podobě uživatelského jména a hesla.

Webová služba bude vyvinuta pomocí WCF technologie od společnosti Microsoft, která využívá aplikační rámec .NET 4.0.

### 5.1 Použité technologie

Tato podkapitola popisuje vybrané technologie pro tvorbu integrační aplikace. Nejdříve je popsána technologie WCF, ve které bude celá aplikace vyvíjena. Je zde popsána struktura WCF technologie a její hlavní výhody.

#### 5.1.1 Windows Communication Foundation

Pro vývoj SOA (Servisně orientovaná architektura) aplikací je na výběr několik technologií, pomocí kterých může aplikace komunikovat s okolím. Programátor ale musí vyvíjet aplikaci tak, aby odpovídala zvolené technologii pro komunikaci. Mezi technologie využívané pro komunikaci SOA aplikací patří mimo jiné následující [10]:

- ASP.NET Web Services
- Web Services Enhancement (WSE)
- .NET Remoting
- Enterprise Services
- Microsoft Message Queuing (MSMQ)

Některé z nich byly popsány v předchozích kapitolách této práce. Tyto technologie mají společné nevýhody, mezi které patří vysoká závislost vývoje dané aplikace na zvolené technologii a vzájemná nekompatibilita jednotlivých technologií mezi sebou.

WCF je jednotný aplikační rámeček, který sjednocuje všechny předchozí technologie od společnosti Microsoft. To znamená, že vývojář nemusí znát programový model technologie, kterou chce využít, ale stačí mu znát koncepci WCF. WCF tak zjednodušuje programátorskou práci a nabízí mnoho dalších rozšiřujících vlastností, mezi které patří například zabezpečení na transportní vrstvě, šifrování zpráv, autentizace uživatele, definice různých protokolů pro připojení k webové službě a jiné.

Základní pojmy spojené s technologií WCF:

**Služba** (*service*) – je základním prvkem WCF aplikace. Jedná se o systém poskytující jeden nebo více koncových bodů (*endpoint*) sloužících pro příjem a odesílání SOAP zpráv. Mimo to služba veřejně poskytuje metadata, která ji popisují. Každá služba se skládá ze tří částí. První část tvoří třída služby (samotná implementace), druhou částí je hostovací prostředí neboli místo, kde služba běží, a poslední částí je jeden nebo více koncových bodů. [10]

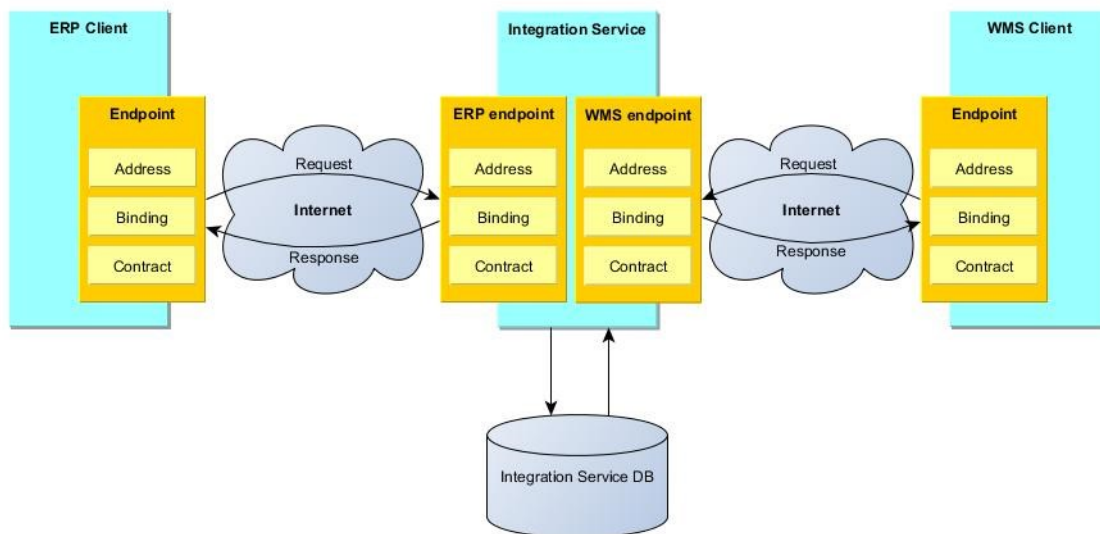
**Koncový bod** (*endpoint*) – místo ve službě, které slouží k přijímání a odesílání zpráv. Koncový bod je tvořen třemi částmi *address*, *binding* a *contract*. Adresa určuje umístění služby neboli adresu, kam budou zasílané zprávy. *Binding* určuje, jakým způsobem bude aplikace komunikovat. Mezi základní informace patří zvolený komunikační protokol, kódování zprávy, zabezpečení, výběr transakcí apod. *Contract* specifikuje rozhraní, které služba poskytuje, její metody a další. *Contract* je nezávislý na adrese a na způsobu komunikace. [10]

**SOAP zprávy** – SOAP zprávy jsou podrobněji popsány v kapitole 3.3.1. WCF aplikace podporují 3 modely posílání zpráv. Prvním typem je [10]:

- *One way* – jedná se o jednosměrnou komunikaci, kdy klient odešle požadavek službě a neočekává odpověď.
- *Request-response* – klient odešle požadavek na webovou službu a čeká na odpověď
- *Duplex* – je obousměrná komunikace mezi klientem a webovou službou. To znamená, že webová služba může vyvolat i nějakou metodu klienta.

**Metadata** - slouží k popisu služby. Metadata obsahují všechny důležité údaje potřebné k tomu, aby na jejich základě mohl být nakonfigurován klient. Díky tomu klient ví, na jakém protokolu služba běží, na jaké adrese atd. [10]

Na obrázku 9 je znázorněn způsob předávání dat mezi systémy ERP a WMS pomocí WCF integrační aplikace, která si jednotlivá data z obou systémů ukládá do lokální databáze a následně je na vyžádání přeposílá cílové aplikaci.

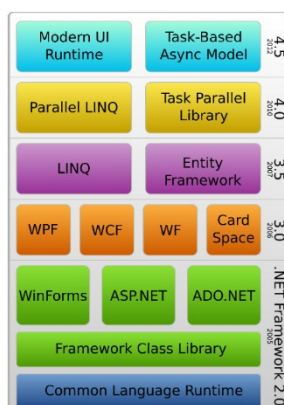


Obrázek 9 - Zobrazení komunikace systémů ERP a WMS se službou

### 5.1.2 Aplikační rámec .NET

Aplikační rámec .NET je určen primárně pro operační systém Microsoft Windows, kde jsou jeho jednotlivé verze součástí Windows. Čtvrtá verze byla vyvíjena společně s vývojovým prostředím Visual Studio 2010 a je doporučenou součástí operačního systému Windows 7. Obsahuje obsáhlé knihovny, které poskytují otestované a znovupoužitelné kódy pro všechny hlavní oblasti vývoje aplikace. Čtvrtá verze aplikačního rámce .NET obsahuje knihovny pro tvorbu WCF aplikací a Entity Framework, který je použit pro mapování relační databáze na objekty.

Na obrázku 10 je zobrazena struktura .NET aplikačního rámce a její jednotlivé knihovny, včetně jejich postupného vývoje od .NET verze 2, která vznikla v roce 2005 až po současnou nejnovější verzi 4.5, která se vyvíjí již od roku 2012 a je standardní součástí Visual Studia 2013 a Windows 8. Verze 4.5 je zpětně kompatibilní s verzí 4.



Obrázek 10 - struktura .NET aplikačního rámce [http://dotnetframeworkdifference.blogspot.cz/]

### 5.1.3 Aplikační rámec Entity Framework

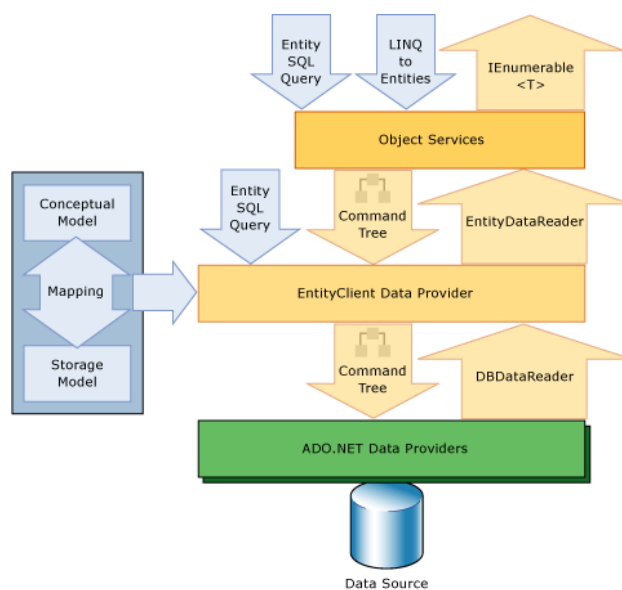
Tento aplikační rámec, vyvinutý společností Microsoft, slouží pro objektově relační mapování SQL databáze. Umožňuje jednoduchý přístup do databáze, převádí data na objekty, se kterými lze libovolně pracovat. Při práci s objekty zůstává v pozadí aplikace a sleduje změny. V dnešní době je tento aplikační rámec doporučován pro práci s databázemi přímo od společnosti Microsoft.

Entity Framework je rozšířením technologie ADO.NET. ADO.NET technologie poskytuje vývojáři přístup k různým datovým zdrojům. To také patří mezi hlavní výhody použití aplikačního rámce Entity Framework. Entity Framework využívá stejné zprostředkovatele operací nad konkrétní databázi (data providery) jako ADO.NET.

Entity Framework lze využít ve třech různých situacích. První situací je, když má vývojář k dispozici existující databázi nebo pokud vývojář začíná návrhem databáze před implementací dalších částí aplikace. Druhou situací je zaměření se na doménovou strukturu aplikace a následné vygenerování databáze z doménových tříd. Posledním scénářem je návrh databázového schéma pomocí vizuálního designeru a následné vytvoření databáze a odpovídajících tříd. Při implementaci integrační aplikace bude vhodné využít první možnost použití, tudíž navrhnout a vytvořit databázi vhodnou pro integraci obou systémů a poté vygenerovat třídy mapující tuto databázi. Pro spolupráci s WCF technologií je nutné generované třídy přizpůsobit tak, aby vyhovovaly požadavkům WCF technologie.

#### Architektura aplikačního rámce Entity Framework

Na obrázku 11 je znázorněna architektura aplikačního rámce Entity Framework a skladba jeho komponent:



Obrázek 11 - Architektura Entity Frameworku [16]

EDM (Entity Data Model) se skládá ze tří hlavních částí - konceptuálního modelu (angl. „Conceptual Model“), mapování (angl. „Mapping“) a databázového modelu (angl. „Storage Model“). Konceptuální model obsahuje třídy a jejich vzájemné vazby, které jsou nezávislé na databázové struktuře tabulek. Databázový model obsahuje tabulky, pohledy, uložené procedury, klíče a vazby jednotlivých tabulek. Mapování uchovává informace o tom, jak je konceptuální model mapován na databázový model. Další komponenty Entity Frameworku se vztahují na dotazování, získávání, ukládání a editaci dat do databáze. [16]

LINQ to Entities je moderní dotazovací jazyk užívaný pro tvorbu dotazů na objektový model. Vrací entity definované v konceptuálním modelu. Dalším dotazovacím jazykem je Entity SQL, který nabízí oproti LINQ to Entities více prostředků, ale je složitější na použití. [16]

Object Service je hlavní částí pro přístup k datům v databázi a pro vrácení hodnot, které z ní byly získány. Tato komponenta je zodpovědná za proces konverze dat získaných z Entity Client Data Provideru do objektové struktury. Hlavní úlohou Entity Client Data Provideru je konvertovat dotazy z LINQ to Entities a Entity SQL do klasického SQL dotazu, který je srozumitelný pro databázi. Komunikuje s ADO.NET providery, které zasílají a přijímají data z databáze. [16]

Díky této architektuře je možné snadněji přistupovat k datům uloženým v databázi, a jelikož se jedná o volně dostupnou technologii, je možné najít velké množství providerů od třetích stran pro připojení různých typů databází.

#### **5.1.4 Návrhový vzor Repository**

Úkolem návrhového vzoru *Repository* je například provádění operací nad daty v databázi. Jedná se o třídy pracující s jedním typem entity. Takže pro každou tabulku v databázi existuje třída *Repository*, která umožňuje vyhledávání, vkládání, mazání a úpravu dat v dané tabulce. Tím je oddělena vrstva přístupu k databázi od logické části aplikace. Proto je možné kdykoliv nahradit databázi za jiný typ bez nutnosti upravovat vrstvu modelu. Při změně se upraví pouze třídy *Repository* daných objektů.

Návrhový vzor *Repository* odděluje doménové objekty od kódu za účelem přístupu k databázi. Tento návrhový vzor se používá při velkém množství doménových tříd a pomáhá zamezovat výskytu duplicitních dotazů na databázi. Jedná se o mezivrstvu při přístupu k databázi, která si uchovává v paměti entity, které potom odesílá do databáze ke zpracování.[17]

## **5.2 Třídní diagram integrační aplikace**

Následující třídní diagram na obrázku 12 zobrazuje strukturu aplikace a její reprezentující tabulkovou strukturu databáze integrační aplikace včetně jejich vazeb. Ke každé třídě reprezentující tabulku databáze existuje třída *repository*, která obsahuje všechny metody pro práci

s databázovými objekty. Třídy repository obsahují metody pro vložení, úpravu, smazání dat v databázi a metody pro získání dat pomocí dotazů na databázi (LINQ to Entities).

V aplikaci jsou implementována dvě rozhraní definující metody dané webové služby pro určitého klienta. První rozhraní definuje metody určené pro ERP klienta a druhé definuje metody pro WMS klienta. Logiku včetně řízení komunikace mezi systémy ERP a WMS obsahuje třída implementující obě výše zmíněná rozhraní. Tato třída také spravuje databázi pomocí tříd repository.





### 5.3 Popis implementace zpracování požadavku skladové operace

Díky použití WCF technologie jsou webové služby volány pomocí SOAP zpráv. Tyto SOAP zprávy obsahují veškeré potřebné informace ohledně autentizace klienta, volaných metod, jejich parametrů a návratových hodnot. Všechny příklady SOAP zpráv jsou výsledkem komunikace mezi systémem ERP s webovou službou a následně komunikace systému WMS s webovou službou. Pro přehlednost jsou v této podkapitole uvedeny pouze příklady SOAP zpráv při zpracování požadavku na zaskladnění. Odesílaná data jsou také zjednodušena kvůli přehlednosti. Všechny tabulky integrační aplikace jsou rozšířeny o atributy stav zprávy a jednoznačné číslo zprávy, které zajišťují správnost přenosu všech odesílaných dat z webové služby.

Příklad SOAP zprávy odeslané systémem ERP do webové služby, která zajišťuje odeslání požadavku na zaskladnění do systému WMS, kde jsou v těle zprávy obsaženy údaje o hlavičce zaskladnění s následným výčtem řádků zaskladnění:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <To s:mustUnderstand="1"
      xmlns="http://schemas.microsoft.com/ws/2005/05/addressing/none">
      http://localhost:50900/ERP_WMS_IntegrationService.svc
    </To>
    <Action s:mustUnderstand="1"
      xmlns="http://schemas.microsoft.com/ws/2005/05/addressing/none">
      http://tempuri.org/IERP_IntegrationService/SendPutAwayToWMS
    </Action>
  </s:Header>
  <s:Body>
    <SendPutAwayToWMS xmlns="http://tempuri.org/">
      <putAwayHeader
        xmlns:a="http://schemas.datacontract.org/2004/07/ERP_WMS_IntegrationService.
        Model" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:CompanyCode>CZ</a:CompanyCode>
        <a:GoodReceiveDate>2015-04-21T09:00:00</a:GoodReceiveDate>
        <a:IncomDocNo>ZS000011</a:IncomDocNo>
        <a:LinesCount>1</a:LinesCount>
        <a:SourceDocNo>106030</a:SourceDocNo>
        <a:SourceDocType>Nákupní objednávka</a:SourceDocType>
        <a:WarehouseCode>BÍLÝ</a:WarehouseCode>
      </putAwayHeader>
      <putawayLines
        xmlns:a="http://schemas.datacontract.org/2004/07/ERP_WMS_IntegrationService.
        Model" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:PutawayLine>
          <a:CompanyCode>CRONUS CZ s.r.o.</a:CompanyCode>
          <a:ExpirationDate i:nil="true"></a:ExpirationDate>
          <a:FromBin>B-08-0001</a:FromBin>
          <a:FromZone>PŘÍJEM</a:FromZone>
          <a:IncomDocCode>ZS000011</a:IncomDocCode>
          <a:IncomDocLineCode>10000</a:IncomDocLineCode>
          <a:ItemCode>1251</a:ItemCode>
          <a:PackageCode i:nil="true"></a:PackageCode>
          <a:ProductionDate i:nil="true"></a:ProductionDate>
          <a:QtyReceived>0</a:QtyReceived>
        </a:PutawayLine>
      </putawayLines>
    </SendPutAwayToWMS>
  </s:Body>
</s:Envelope>
```

```

    <a:QtyToReceive>10</a:QtyToReceive>
    <a:ReceiptDocLineNo>10000</a:ReceiptDocLineNo>
    <a:ReceiptDocNo>P_000013</a:ReceiptDocNo>
    <a:SenderNo>01905382</a:SenderNo>
    <a:SenderType>Dodavatel</a:SenderType>
    <a:ToBin>B-04-0005</a:ToBin>
    <a:ToZone>VYSKL</a:ToZone>
    <a:UnitOfMeasureCode>KS</a:UnitOfMeasureCode>
    <a:Urgent">False</a:Urgent">
    <a:WarehouseCode>BÍLÝ</a:WarehouseCode>
  </a:PutawayLine>
</putawayLines>
</SendPutAwayToWMS>
</s:Body>
</s:Envelope>

```

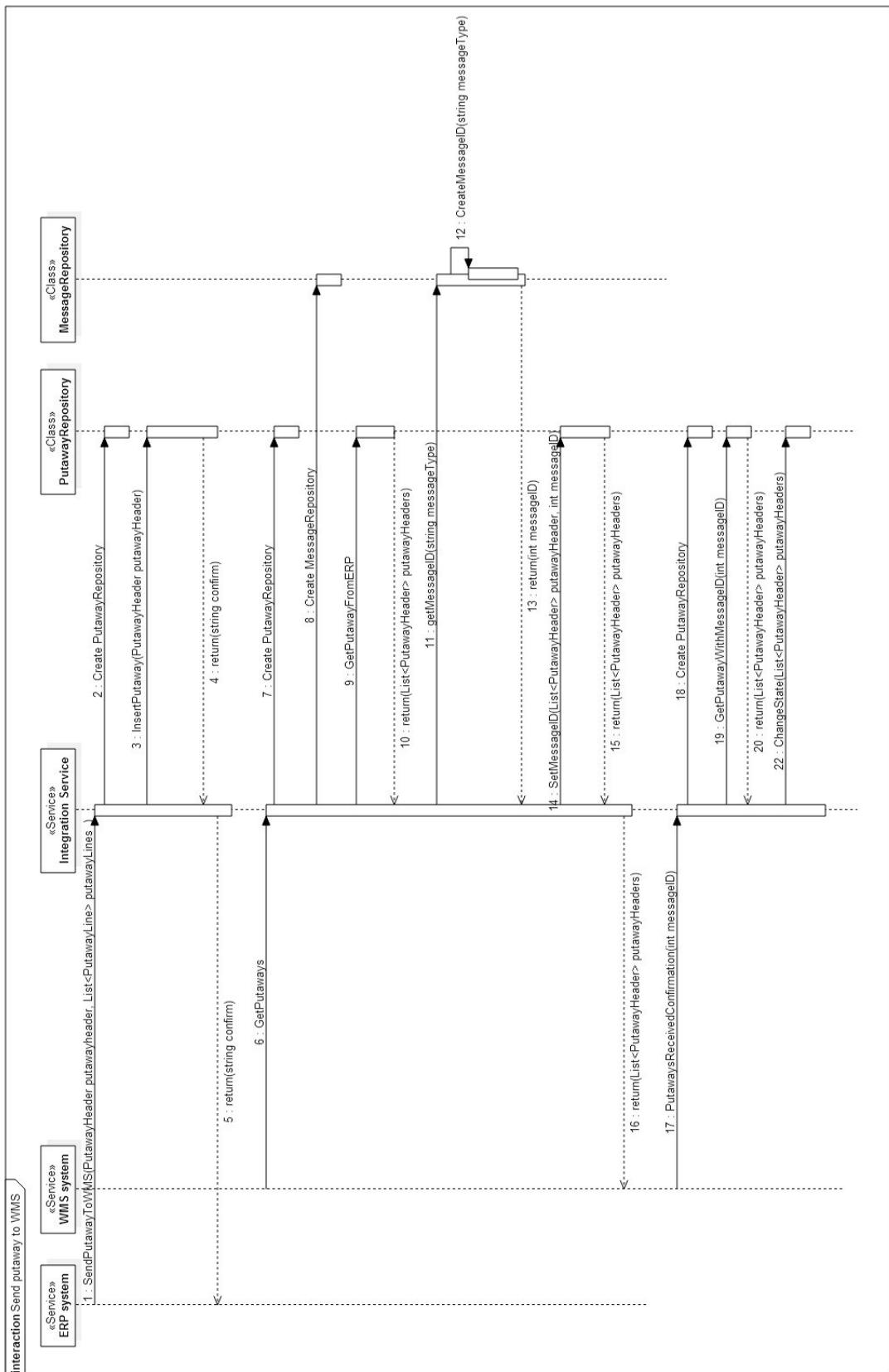
Webová služba po přijetí dokladu zaskladnění ze systému ERP upraví v přijaté hlavičce (*PutawayHeader*) atribut stav zprávy na „k odeslání“. Tento příznak určuje, že má být doklad zaskladnění odeslán do aplikace WMS. Po přijetí dokladu zaskladnění webovou službou je odeslána odpověď s textovou konstantou označující úspěšnost zpracování dokladu zaskladnění původního požadavku (např. příznakem dokončeno).

Následně jsou na vyžádání odeslány všechny doklady zaskladnění, se stavem zprávy „k odeslání“, odeslány systému WMS. V integrační aplikaci musí být před odesláním odpovědi vygenerováno a přiřazeno jednoznačné číslo zprávy všem posílaným dokladům zaskladnění.

Posledním krokem odeslání požadavku na zaskladnění z ERP systému do systému WMS je potvrzení přijetí dokladů zaskladnění systémem WMS z webové služby. Systém WMS odešle SOAP zprávu obsahující parametr s jednoznačným číslem zpráv přijatých dokladů zaskladnění. Webová služba následně po přijetí jednoznačného čísla dokladu zaskladnění upraví stav zpráv u všech dokladů zaskladnění s daným číslem na „odesláno“.

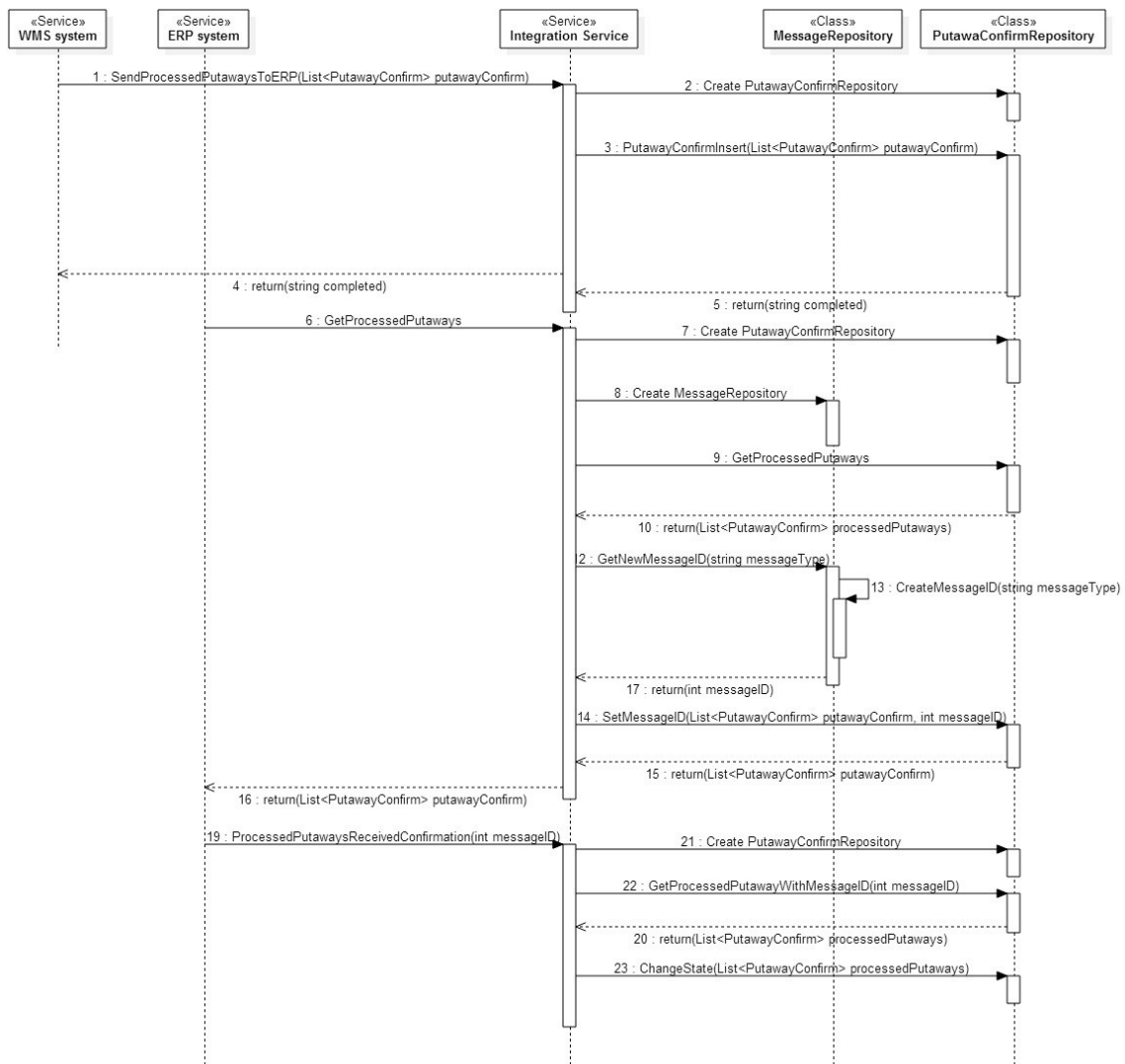
Po přijetí potvrzení o doručení dokladů zaskladnění systémem WMS jsou všechny hlavičky zaskladnění s identifikačním číslem (v tomto případě čtyřicet sedm) označeny za odeslané, aby nedocházelo k opětnému odesílání zpráv.

Sekvenční diagram na obrázku 13 shrnuje průběh odeslání požadavku na zaskladnění ze systému ERP do systému WMS skrze integrační webovou službu. V sekvenčním diagramu jsou znázorněny jednotlivé metody integrační webové služby, které jsou volány systémem ERP a WMS pro následnou výměnu důležitých dat. Dále obsahuje operace vykonané webovou službou pro správu dat.



Obrázek 13 - sekvenční diagram pro odeslání dokladu zaskladnění do systému WMS

Po zpracování zaskladnění systémem WMS probíhá odeslání výsledků neboli zpracovaných řádků zaskladnění zpět systému ERP. Průběh je podobný jako při odesílání požadavku na zaskladnění ze systému ERP do systému WMS. Rozdíl je v přenášených datech. V případě zaskladnění se jedná o jednotlivé řádky zpracovaného zaskladnění, které odpovídají řádkům odeslaných ze systému ERP. Průběh odeslání zpracovaných řádků je znázorněn sekvenčním diagramem na obrázku 14.



Obrázek 14 - Sekvenční diagram potvrzení zaskladnění ze systému WMS

Stejným způsobem jako při požadavcích zaskladnění probíhá komunikace mezi ERP a WMS v případě požadavků vyskladnění, kdy se odesílají požadavky z ERP, následně musí být potvrzeno přijetí požadavků zaskladnění systémem WMS a nakonec musí být výsledky zaskladnění odeslány systémem WMS do systému ERP. Přijetí výsledků zaskladnění musí opět systém ERP potvrdit.

Rozdílnou situací může být odesílání hlavních dat, jakou jsou informace o zboží, zákaznících, dodavatelích apod., kdy je směr určen pouze směrem z ERP systému, který udává a spravuje tato data, do systému WMS. Ten má metody pouze pro přijetí nových nebo upravených hlavních dat. Stále přitom platí nutnost potvrzení přijatých dat pomocí jednoznačného čísla zprávy.

## **5.4 Implementace klienta webové služby**

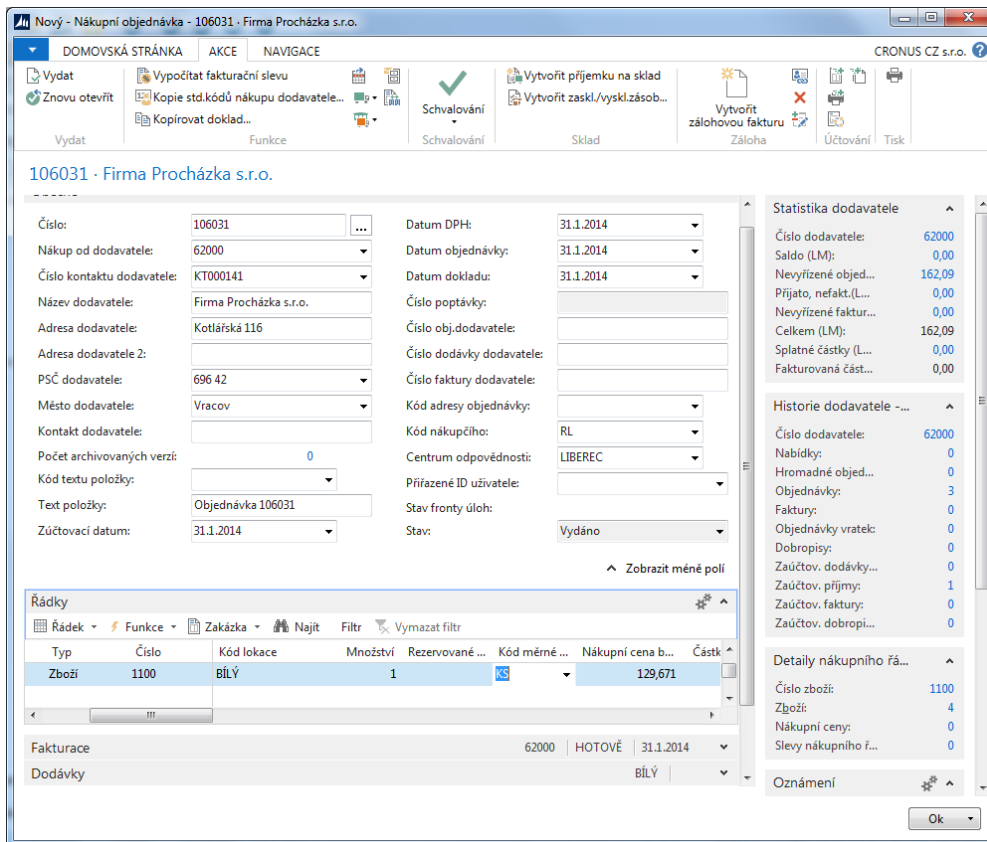
Integrační aplikace WCF byla implementována pomocí návrhu zmíněného výše. Tato kapitola se zaměřuje na popis implementace a ověření funkčnosti na straně klienta. Klientem je ERP aplikace Microsoft Dynamics NAV 2013 druhého vydání. Na straně WMS je nutná implementace od dodavatele softwaru pro řízené sklady.

Dynamics NAV 2013 R2 podporuje práci s .NET knihovnami, což usnadňuje práci programátorům. Za účelem odesílání dat do webové služby byla vygenerována knihovna klienta dané webové služby a poté naimportována do programu Dynamics NAV. Takto vygenerovaná knihovna obsahuje všechny důležité informace ohledně připojení a metod, které webová služba nabízí včetně datových typů, které zpracovává.

Pro práci s webovou službou byla vytvořena nová instance serveru Dynamics NAV pod názvem NAS. Tato instance je nastavena, aby se spouštěla ihned po spuštění serveru a v nastavených časových intervalech volala metody webové služby. Například pro odesílání dokladů zaskladnění je nastaveno časové rozmezí jedné minuty, kdy se opakují požadavky na odeslání označených dokladů zaskladnění. Stejný čas (jedné minuty) je nastavený i na opakování požadavků pro příjem zpracovaných dokladů zaskladnění ze systému WMS.

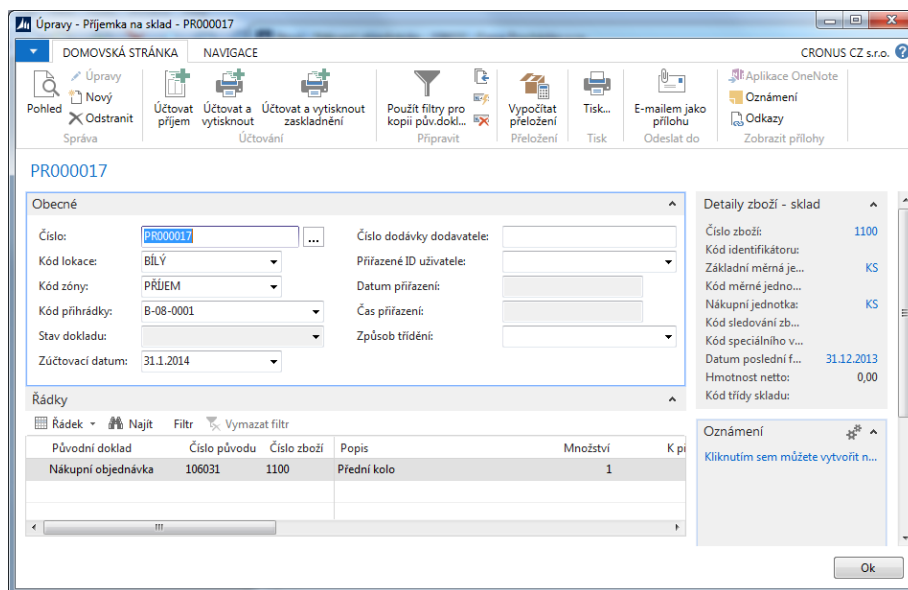
U hlavních dat, kterými jsou informace o zákazníkovi, dodavateli, zboží a jiných, jsou webové služby volány při každé úpravě potřebných polí na daných záznamech nebo při vytvoření nové položky.

Pro správnou funkčnost byla provedena úprava v systému ERP, kde byla tabulka aktivity skladu rozšířena o pole s předdefinovanými hodnotami, pomocí kterého jsou označeny řádky aktivity skladu k odeslání do WMS. Jako první je v ERP systému Dynamics NAV vytvořena nákupní objednávka (obrázek 15).



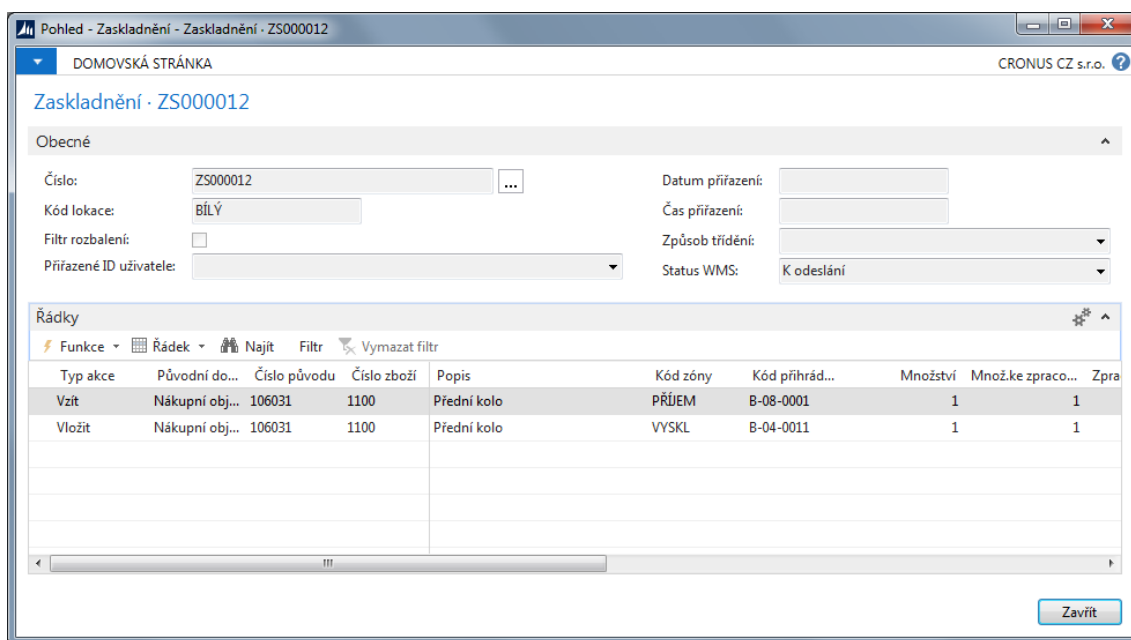
Obrázek 15 - okno Dynamics NAV pro vytvoření nákupní objednávky

Po vydání nákupní objednávky může být vytvořena příjemka na sklad v menu „Akce“ položka „Vytvořit příjemku na sklad“ zobrazená na obrázku 16.



Obrázek 16 - Příjemka na sklad v systému Dynamics NAV

Po zaúčtování příjemky na sklad, které lze vyvolat pomocí funkce „Účtovat příjem“, je vytvořen doklad zaskladnění obsahující manipulaci se zbožím v rámci skladu. V hlavičce dokladu zaskladnění je zobrazené nové přidané pole WMS status, které bylo automaticky nastaveno po vytvoření na stav „K odeslání“. Doklad zaskladnění vytvořený v Dynamics NAV je zobrazen na obrázku 17.



Obrázek 17 - doklad zaskladnění v systému Dynamics NAV

Po odeslání dokladu zaskladnění do systému WMS je pole status WMS nastaveno na hodnotu Odesláno a systém ERP čeká na přijetí zpracovaných řádků zaskladnění ze systému WMS. Poté je provedena kontrola množství, které bylo zaskladněno systémem WMS, a pokud nezůstává již žádné zboží k zaskladnění, je doklad registrován. Tím se zboží stává disponibilním pro jeho další zpracování, například pro prodej a jiné.

## 5.5 Rizika a limity navrženého rozhraní

Integrační aplikace předpokládá, že příjem zboží na sklad a výdej zboží ze skladu probíhá na základě dokladů zaskladnění a vyskladnění ze systému ERP. Pokud systém ERP nemá možnost vytvoření dokladů zaskladnění a vyskladnění, jsou jednotlivá data pro odeslání požadavku do integrační aplikace plněna z nadřazených dokladů, kterými jsou příjem na sklad nebo dodávka ze skladu.

Důležitou součástí komunikace mezi systémy ERP a WMS je také synchronizace čtyř základních zón skladu. Jedná se o nejdůležitější zóny skladu, mezi které patří zóna příjmu, zóna vyskladnění (zásob), zóna přeložení a dodací zóna. Pomocí těchto zón určuje ERP systém následně zpracování



zboží ve skladu. Například pokud existuje prodejní objednávka na zboží, které není disponibilní na daném skladu, je vytvořena rezervace, která určuje, že při nákupu zboží na sklad má být určité množství přesunuto na výdej zákazníkovi. V takovém případě jsou v dokladu zaskladnění vytvořeny dva řádky. První řádek označuje přesun celkového množství zboží poníženého o množství zboží v zákaznických rezervacích z příjímací zóny do vyskladňovací zóny. Druhý řádek označuje přesun rezervovaného množství zboží z příjímací zóny do zóny přeložení. Následně je vytvořen doklad vyskladnění na dané zboží označující přesun ze zóny přeložení do dodací zóny.

Protože je aplikace postavena na technologii WCF, která zahrnuje webové služby, je závislá na stálém připojení k internetu. Pokud dojde k výpadku spojení, funkčnost jednotlivých systémů to nijak neovlivní, pouze se při opětovném připojení k internetu odešlou všechny doklady generované při výpadku. SOAP zpráva má ve WCF aplikaci definovanou maximální velikost, která je pro běžný provoz dostatečně velká, aby nedocházelo k chybám kvůli velikosti předávaných zpráv. Pokud by bylo webových služeb využito pro prvotní synchronizaci například hlavních dat mezi systémy ERP a WMS, musí být ohlídána maximální velikost SOAP zpráv a v případě velkého množství dat, rozdělit přenos dat do více požadavků.

Pro případ kontroly dat v integrační aplikaci je určen uživatel, který má přístup do databáze integrační aplikace. Aplikace je navržena tak, že jsou data ihned po zpracování a odeslání smazána, aby nedocházelo k zaplňování místa na serveru. Při výskytu případných problémů je potom nutná přímá komunikace mezi systémy ERP a WMS a odeslání opravných dokladů. Řešením do budoucna může být vytvoření záložní databáze, kam budou odesílána zpracovaná data pro budoucí kontrolu.

## 6 Závěr

Důležitou součástí podnikových systémů je integrace s dalšími informačními systémy společnosti, která má za úkol zajistit korektnost dat a jejich přenos v reálném čase mezi jednotlivými systémy. Tato práce je zaměřena na integraci systémů ERP a WMS. Za účelem správného návrhu univerzálního rozhraní pro komunikaci mezi těmito systémy bylo nutné podrobněji nastudovat a popsat funkčnost, strukturu a využití jednotlivých systémů. Veškeré získané informace jsou popsány v druhé kapitole této práce.

Další důležitou částí této diplomové práce bylo studium middleware neboli systémového prostředí, které umožňuje procesům, programům a aplikacím vzájemnou interakci a komunikaci v distribuovaném systému prostřednictvím sítě. Toto prostředí je podrobněji popsáno ve třetí kapitole této práce včetně jeho architektury a rozdělení. Tato kapitola také obsahuje základní představení integračních nástrojů reprezentující různé druhy integrace mezi aplikacemi.

Ve čtvrté kapitole je provedena analýza požadavků skladových operací společných pro systémy ERP a WMS. V této analýze je výčet skladových operací včetně vzájemné komunikace mezi systémy ERP a WMS a nastínění důležitých dokladů pro přenos informací. Díky této analýze bylo možné určit hranice komunikace mezi systémy a předávané doklady potřebné pro skladové operace.

V poslední části této práce je popsán návrh, implementace a funkčnost integrační aplikace pro vzájemnou komunikaci mezi systémy ERP a WMS. Důležitou součástí implementace byla volba technologie, na které je tato integrační aplikace postavena. Pro implementaci byla zvolena technologie WCF kvůli jejímu univerzálnímu použití, které zajišťuje komunikaci napříč různými platformami. Tato technologie sjednocuje mnoho integračních technologií od společnosti Microsoft a snaží se využít jejich výhod. Její výhodou je usnadnění vývoje aplikace, které ale předpokládá hlubší znalost dané technologie.

Pro implementaci integrační aplikace bylo použito vývojového prostředí Visual Studio 2012. Tato integrační aplikace obsahuje vlastní databázi pro uchování přenášených dat ze systému ERP do systému WMS a naopak. Celá integrační aplikace byla vyvíjena na technologii WCF a je připravena pro nasazení na ISS lokálního serveru. Pro testovací účely byl nainstalován Software Microsoft Dynamics NAV 2013 R2 s licencí firmy Allium s.r.o., který byl následně propojen s integrační aplikací. V programu Dynamics NAV 2013 byla vytvořena *codeunita* (objekt obsahující programovou část), která zajišťuje komunikaci s integrační aplikací a odesílá jí potřebná data.

Při testování byla vypnuta všechna zabezpečení, která aplikace WCF poskytuje za účelem sledování komunikace a odesílaných dat pomocí SOAP zpráv. V ostrém provozu se předpokládá

se zabezpečením transportní vrstvy pomocí HTTPS protokolu a šifrování zpráv, které je možné nastavit pomocí konfiguračního souboru integrační aplikace.

Výhodou navržené integrační aplikace je možnost výměny některého ze systémů ERP nebo WMS při zachování funkčnosti druhého nevyměněného systému. To znamená, že pokud je například systém WMS nahrazen za jiný produkt WMS, je potřeba implementovat úpravy pouze v nově nahrazeném WMS systému zatímco v ERP systému nejsou zapotřebí žádné změny. To stejné platí i v případě výměny ERP systému, kdy není potřeba upravovat funkcionalitu WMS systému. Další výhodou tohoto řešení je například integrace více různých systémů pro řízení sklady do jednoho ERP systému.

Díky této práci jsem si prohloubil znalosti ohledně skladových operací v běžném provozu a řešení obecných skladových požadavků, dále jsem se při implementaci seznámil s integračními aplikacemi a moderní technologií WCF. Ačkoliv existují integrační nástroje pro integraci různých systémů, tato práce byla zaměřena na specifitější integraci systémů ERP a WMS, které jsou nejčastěji používány v distribučních organizacích. Pro tyto účely není nutné využívat složité integrační nástroje, které je nutné nahradit při výměně technologie jednotlivých systémů. Navržená aplikace poskytuje komunikaci mezi systémy při dodržení jednoduchosti implementace v systémech ERP a WMS.

## 7 Použitá literatura

- [1] LUSZCZAK, Andreas a Robert SINGER. *Microsoft dynamics NAV: výukový kurz*. Vyd. 1. Brno: Computer Press, 2011, 373 s. ISBN 9788025128510
- [2] FRICK, D. *SAP R/3 kompletní průvodce*. Brno: Computer Press, 2007. 733 s. ISBN 978-80-251-1750-7.
- [3] *Skladové hospodářství v ERP versus systém řízených skladů WMS*. [online]. Dostupné z: <http://www.aimagazine.cz/cs/tema/842-skladove-hospodarstvi-v-erp-versus-system-rizenych-skladu-wms>
- [4] EMMETT, Stuart. *Řízení zásob: jak minimalizovat náklady a maximalizovat hodnotu*. Vyd. 1. Brno: Computer Press, 2008, vi, 298 s. Praxe manažera (Computer Press). ISBN 978-80-251-1828-3.
- [5] GÁLA, L., POUR, J., ŠEDIVÁ, Z. *Podniková informatika*. 2.vyd. Praha: Grada, 2009. 496s. ISBN 978-80-247-2615-1
- [6] BOX, Don. A *Brief History of SOAP*. O'Reilly XML.com. [online]. Dostupné z: <http://www.xml.com/pub/a/2001/04/04/soap.html>.
- [7] BOX, Don, EHNEBUSKE, David, KAKIVAYA, Gopal, LAYMAN Andrew, MENDELSON, Noah, NIELSEN, Henrik Frystyk, THATTE, Satish, WINER, Dave. *Simple Object Access Protocol (SOAP) 1.1*. W3C Note. 2000. Dostupné z: <http://www.w3.org/TR/SOAP/>.
- [8] MITRA, Nilo a Yves LAFON. SOAP Version 1.2 Part 0. W3C. *W3C Recommendation* [online]. 2007. Dostupné z: <http://www.w3.org/TR/soap12-part0/>
- [9] CHRISTENCEN, Francisco CURBERA, Greg MEREDITH a WEERAWARANA. *Web Services Description Language (WSDL) 1.1* [online]. 2001. Dostupné z: <http://www.w3.org/TR/wsdl>
- [10] LARYŠ, Kryštof. *WCF PRO ZAČÁTEČNÍKY: 1. DÍL: TEORIE, ZÁKLADNÍ POJMY* [online]. 2009. Dostupné z: <http://www.netstudent.cz/Articles/wcf-pro-zacatecniky-1-dil-teorie-zakladni-pojmy>
- [11] HOLLOWAY, Simon. *BPM and SOA, Cordys style* [online]. 2008. Dostupné z: <http://www.it-director.com/technology/content.php?cid=10295>

- [12] HOHPE, Gregor, WOOLF, Bobby. Enterprise integration patterns: designing, building, and deploying messaging solutions. San Francisco: Addison-Wesley, c2004, li, 683 s. Addison-Wesley signature series. ISBN 0321200683.
- [13] ROUSE, Margaret. Microsoft SSIS (SQL Server Integration Services). [online]. [cit. 2015-04-29]. Dostupné z: <http://searchsqlserver.techtarget.com/definition/Microsoft-SSIS-SQL-Server-Integration-Services>
- [14] ŠTUMPF, Jindřich. *Integrace (mezi)podnikových aplikací: Integrace pomocí SonicMQ - výměny zpráv dle standardu JMS*. [online]. Dostupné z: <http://www.systemonline.cz/clanky/integrace-mezi-podnikovych-aplikaci.htm>
- [15] LHEUREUX, B., SCHULTE, R., NATIS, I., MCCOY, D., GASSMAN, B., SINUR, J., THOMPSON, J., PEZZINY, M., KENNEY, F., FRIEDMAN, T., GILBERT, M., PHIER, G. *Gartner RAS Core Strategic Analysis Report R-22-2153*. 2004. [online]. Dostupné z: <http://www-01.ibm.com/software/info/websphere/partners4/articles/gartner/garwho.html#1.1>
- [16] MICROSOFT. 2013. *Entity Framework Overview* [online]. Dostupné z: [https://msdn.microsoft.com/cs-cz/library/bb399567\(v=vs.110\).aspx](https://msdn.microsoft.com/cs-cz/library/bb399567(v=vs.110).aspx)
- [17] SPROTT, David a WILKES. 2004. *Understanding Service-Oriented Architecture* [online]. Dostupné z: <https://msdn.microsoft.com/en-us/library/aa480021.aspx>

## 8 Přílohy

### 1. Obsah CD

1. Text diplomové práce ve formátu PDF/A
2. Spustitelný program
  - a. soubor ERP\_WMS\_IntegrationService.sln spustitelný ve vývojovém prostředí Visual Studio 2012
  - b. databáze potřebná pro běh integrační aplikace
  - c. zdrojové kódy aplikace