

**Česká zemědělská univerzita v Praze**



**Diplomová práce**

**Lukáš Němec**

**Elektronická zdravotní dokumentace**

Katedra informačního inženýrství

Vedoucí diplomové práce: prof. RNDr. Jiří Vaníček CSc. a Ing. Marek Pícka,  
katedra informačního inženýrství

2009

## **Poděkování**

Děkuji panu profesoru RNDr. Jiřímu Vaníčkovi CSc. a panu Ing. Markovi Píckovi za hodnotné rady a odborné vedení během vypracovávání obsahu mé práce.

## **Prohlášení**

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů a vlastní praxe.

V Praze dne 8. dubna 2009

.....

Lukáš Němec

# Elektronická zdravotní dokumentace

## Souhrn

Tato práce se zabývá problematikou elektronické zdravotní dokumentace, v ohledu na stávající prostředí v rámci České republiky. Současně se i věnuje nástinu vhodného rozvoje, respektive vývoje používaných standardů a technologií pro zajištění výměny a zpracovávání informací mezinárodním rozsahu.

Práce popisuje architekturu celého řešení a jednotlivé části systému. Dále jsou zmíněny způsoby napojení na další systémy v rámci zdravotnictví. V neposlední řadě je přiblížen vývoj části takového systému.

## Klíčová slova

Elektronická zdravotní dokumentace, datový standard, komunikační standard, Webové služby, DASTA, HL7, WS

# Electronic Health Record

## Summary

This work deals with electronic medical records matters, in the current environment in the Czech Republic. At the same time, and pays an outline of an appropriate development of standards and technologies used to ensure the exchange of information processing, and international scale.

The work describes the architecture of the entire solution and the individual components of the system. There are also mentioned ways connect with other systems within the health sector. Last but not least is the approach the development of such a system.

## Keywords

Electronic health record, data standard, communication standard, Web services, DASTA, HL7, WS

## Obsah

1 Úvod .....	4
2 Cíl práce a metodika .....	5
3 Pojem Elektronické zdravotní dokumentace .....	7
3.1 Cíl Elektronické zdravotní dokumentace .....	8
3.2 Koncept EZD .....	8
4 Standardy a komunikace v rámci EZD .....	11
4.1 Datové standardy .....	11
4.1.1 DASTA.....	12
4.1.1.1 DS 01.01.....	12
4.1.1.2 DS 02.xx.xx.....	13
4.1.1.3 DS 03.xx.xx.....	14
4.1.1.4 DS 04.xx.xx.....	14
4.1.1.4.1 Struktura datového souboru .....	14
4.1.1.4.2 Číselníky NČLP, DS, UZIS .....	19
4.1.2 HL7.....	20
4.1.2.1 HL7 v2.x .....	20
4.1.2.2 HL7 v3 .....	21
4.1.2.3 Struktura standardu HL7 v3.....	21
4.1.2.3.1 Informační modely .....	22
4.1.2.3.2 Statická struktura a atributy .....	27
4.1.2.3.3 Slovníková doména .....	28
4.1.2.4 Srovnání HL7 a DASTA.....	28
4.2 Webové služby .....	29
4.2.1 Stručný popis jednotlivých specifikací.....	31
4.2.1.1 Transportní protokol.....	31
4.2.1.2 Komunikační protokol SOAP .....	31
4.2.1.3 Popis webové služby WSDL.....	32
4.2.1.4 Rozšíření WS: specifikace WS-*.....	33
4.2.2 Zpracování zpráv pomocí WS.....	34
4.2.3 Využití WS v SOA architektuře .....	36
5 Analýza systému EZD.....	39
5.1 Use Case digramy systému EZD .....	39
5.1.1 Přihlášení klienta, zdravotnického pracovníka .....	39
5.1.2 Správa EZD klientem .....	40
5.1.3 Správa EZD zdravotním pracovníkem .....	41
5.1.4 Vstup zdravot. zařízení/pracovníka pomocí B2C brány .....	42
5.1.5 Zápis a nahlížení nad lékařskými zprávami .....	43
5.2 Databázový model aplikace – zdravotní zprávy .....	45
5.2.1 Použité kódové konvence .....	45
5.2.1.1 Tabulky .....	45
5.2.1.2 Sloupce tabulky obecně.....	45
5.2.1.3 Primární klíče.....	46
5.2.1.4 Cizí klíče (referenční integrita) .....	46
5.2.1.5 Indexy .....	46
5.2.1.6 Alternativní klíče .....	47
5.2.1.7 Primární klíče tabulek (indexy).....	47

5.2.2 Lékařské záznamy a zdravotní dokumentace.....	47
5.2.2.1 Lékařské odbornosti .....	48
5.2.2.1.1 Skupiny odborností - [MED_ExpertiseGroup].....	49
5.2.2.1.2 Odbornost - [MED_Expertise].....	50
5.2.2.1.3 Odbornosti ZP a ZZ.....	50
5.2.2.2 Lékařské zprávy a přiložené informace .....	50
5.2.2.2.1 Typy lékařské zprávy - [MED_RecordType] .....	51
5.2.2.2.2 Stav lékařské zprávy.....	52
5.2.2.2.3 Lékařská zpráva .....	53
5.2.2.2.4 Přiložené soubory lékařské zprávy .....	54
5.2.2.2.5 Léky.....	55
6 Ukázka aplikace EZD.....	56
6.1 Architektura brány B2C .....	56
6.1.1 Zabezpečení a autentizace.....	56
6.1.2 Zabezpečení komunikace .....	56
6.1.3 Autentizace uživatelů a požadavků.....	57
6.2 Specifikace služby .....	57
6.2.1 Autentizace požadavků - EZDAuth .....	57
6.2.2 Popis brány.....	58
6.2.3 Datová schránka.....	58
6.2.3.1 Výběr odpovědi dle korelačního identifikátoru .....	59
6.2.3.2 Výběr první zprávy ve schránce.....	59
6.2.3.3 Potvrzení přijetí zprávy .....	60
6.2.4 Klientské služby .....	60
6.2.4.1 Odeslání požadavku .....	60
7 Závěr.....	62
7.1 Přínos práce .....	62
7.2 Co v práci není .....	63
7.3 Budoucnost .....	63
8 Seznam literatury .....	64
8.1 Seznam obrázků .....	65
8.2 Seznam příkladů .....	65
8.3 Seznam tabulek.....	65
9 Přílohy.....	67
9.1 Dokumentace Use Case.....	67
9.2 Seznam zkratk.....	67
9.3 Seznam pojmů .....	68

# 1 Úvod

Žijeme v době, kdy čím dál větší část populace těží z postupné elektronizace vyměňovaných informací, ať se jedná o formu aktivní, kdy člověk je přímo iniciátorem této výměny tak pasivní, kdy je člověku zajištěna obsluha nějakého požadavku, elektronicky, bez jeho vědomí. Tento postupný vývoj je vidět pomalu na všech polích lidské činnosti.

Ve zdravotnictví je jedním z nejdůležitějšího prvku informace o pacientovi, zdrojem těchto informací je zdravotní dokumentace pacienta. Ta je stále ještě vedena velmi často v papírové podobě a uchovávána je povětšinou na místě jejího vzniku, tedy v konkrétním zdravotním zařízení, kde došlo ke zdravotnímu úkonu. Jako taková plní svůj účel vedení informace o pacientu, ale má několik velmi významných nevýhod. Z pohledu práce s informací o pacientu jde hlavně o následující nevýhody:

- Uložení na jednom místě způsobuje špatnou dostupnost pro potřeby ostatních zdravotnických zařízení
- Zdlouhavý způsob hledání požadované informace a její obtížné automatické zpracování
- Náročné zpracování pro různá statistická vyhodnocení
- Nároky na uložení a archivaci

Při bližším pohledu na lékařskou péči, způsoby léčby, určení správné diagnózy je vidět, že léčba konkrétního pacienta je závislá na jeho předchozích či současných zdravotních problémech, různých vrozených dispozic, aktuálně či trvale užívané medikaci a dalších faktorech, které jsou uvedeny ve zdravotní dokumentaci pacienta. Při pohledu na popsané skutečnosti, může špatná dostupnost informací o zdravotním stavu pacienta způsobit jeho možnou špatnou léčbu a v některých případech způsobit zhoršení pacientova stavu, nebo dokonce smrt.

Proto je důležitá elektronizace zdravotní dokumentace, neboť informace uložená v elektronické podobě umožňuje snadnější automatizaci a implementaci vyhledávacích mechanismů, které ve výsledku zrychlí a usnadní přístup k požadované informaci. Součástí elektronického uchovávání zdravotních informací je i volba datového standardu, který popisuje uložené informace. Další nutností je zajištění komunikace mezi jednotlivými účastníky (systémy) ve zdravotnictví. Pro zajištění snadné dostupnosti informace se nabízí jako nejvhodnější centralizované úložiště, které nabízí permanentní možnost přístupu k požadované informaci. Z pohledu implementace se jedná vytvoření jednoho rozhraní pro odesílání a příjem informací v každém systému. Systém takového typu umožní jak získávat zdravotní informace v rámci českého zdravotnictví tak umožní i výměnu na mezinárodní úrovni. Pacient tak získá možnost mít prakticky odkudkoliv přístup ke své zdravotní dokumentaci.

## 2 Cíl práce a metodika

Mým cílem v této práci je snaha přiblížit problematiku tvorby Elektronické zdravotní dokumentace (dále v textu budu uvádět i zkratku EZD), návrh architektury a služeb systému, vývoj a technologie, které je utváří. V práci jsou analyzovány některé neduhy provázející vývoj takového to systému a je nastíněn směr vývoje technologií a služeb, které EZD má nabízet všem zúčastněným stranám.

V první řadě se pokusím provést rozbor standardů pro datový přenos informace s cílem popsat a zjistit jejich rozdíly, výhody a nevýhody a jejich způsob použití pro účely EZD.

Dalším úkolem je analýza řešení EZD jako platformy. Mým hlavním úkolem je vytvoření uceleného pohledu na tento systém. Výstupem je Use Case analýza všech operací vycházející z poznatků v předchozích částech této diplomové práce. Součástí analýzy je i vytvoření metodiky a návrhu datové struktury EZD.

Jednotlivé části práce jsou rozděleny do samostatných tematických kapitol, jejichž obsah odpovídá zvolené metodice a cílům diplomové práce.

V první části práce se snažím nastínit co vlastně EZD znamená, její přínosy a využití ve zdravotnictví jako takovém. Zároveň zde přibližuji určité zažité způsoby nahlížení na zdravotní informace jako takové a rozdíl oproti jejich interpretaci v prostředí EZD. Dále problémy svázané s implementací EZD v prostředí resortu jako je zdravotnictví. Nakonec se věnuji otázce, co přinese systém EZD a pohled na koncept takového systému.

Druhou část práce věnuji přiblížení používaných standardů. Tato kapitola je rozdělena na dvě hlavní podkapitoly. V první podkapitole se soustředím na popis a práci s dvěma druhy datových standardů, konkrétně DASTA a HL7. Je zde popsáno, jak jsou tyto standardy vytvořeny, jak se s nimi pracuje, jejich srovnání a použití. Druhá část je věnována standardu WS, respektive souboru standardů souhrnně označovaných jako webové služby. Tato část se věnuje popisu jednotlivých specifikací, zpracování jednotlivých zpráv pomocí WS a současně použití WS<sup>1</sup> v architektuře SOA<sup>2</sup>.

Třetí částí práce se zabývám analýzou jednoho z možného přístupu k problematice EZD. Jako první se zde věnuji základní analýze z pohledu případu použití, kde popisuji vybrané způsoby práce s aplikací. Blíže se soustředím na způsoby přihlášení, správy EZD a samotným zdravotním zprávám. V této kapitole je uveden pouze určitý nástřel případů použití, kompletní analýza je součástí přílohy diplomové práce. Dále se zde zabývám metodikou návrhu datového modelu z hlediska zvolených konvencí a analýze a vytvoření obecných entit pro úložiště zdravotnické dokumentace.

---

<sup>1</sup> WS – Web Service, webové služby

<sup>2</sup> SOA – Service Oriented Architecture

Čtvrtá část mé práce z části navazuje na předchozí kapitolu, respektive rozpracovává způsob výměny informací různého druhu mezi systémem EZD a externími systémy, za splnění podmínek bezpečností a zároveň jednoduchosti použití.



### 3 Pojem Elektronické zdravotní dokumentace

Pod pojmem EZD se skrývá celá oblast jednotlivých služeb zajišťující přenos a správu informace a návaznost jednotlivých kroků odehrávajících se v rámci zdravotnické péče. Jde o systém spadající pod koncepci eGovernmentu v rámci České republiky, konkrétně by se mělo jednat o jeden z hlavních projektů koncepce eHealth.

V prvé řadě je vhodné přiblížit pojem eHealth (1). Původně v akademické pojetí panoval názor, že je vhodné se tomuto názvosloví vyvarovat jako nepřesnému a zavádějícímu. Nicméně v dnešní době se již dlouho toto označení dané problematiky běžně používá v technické či výzkumné literatuře. Také je třeba zmínit, že pojem přesně vyjadřující danou problematiku neexistuje, protože se jedná o neustále se rozvíjející se oblast.

Spíše než o nějaké vyjádření pouhého technologického vývoje se jedná o mnohem komplexnější termín, vyjadřující myšlenku podle (2), že „eHealth je perspektivní oblast na křižovatce medicínské informatiky, veřejného zdraví a obchodu, odkazující na zdravotní služby a informace doručené a obohacené skrze využití sítě Internet a dalších spřízněných technologií. V nadřazeném smyslu necharakterizuje tento termín pouze technický vývoj, ale současně určitý duševní stav, způsob uvažování, postoj, závazek být co v nejtěsnějším propojení a celkové uvažování pro zkvalitnění lékařské péče lokálně, regionálně a celosvětově za použití informačních a komunikačních technologií.“

Vrátím se zpátky k pojmu EZD, ten totiž celkem přesně reflektuje vysvětlení samotného pojmu eHealth. Jde o to, že technický vývoj zde spočívá spíše ve smyslu použití určitých technologií v prostředí, které je ještě do nedávné doby nevyužívalo. Jde zde samozřejmě i o vývoj technologický, hlavně na poli standardů, ale hlavní snahou a i chtěným přínosem je změna způsobu uvažování nad prací s informacemi v oblasti zdravotnictví. Tato změna ovšem nebude nikdy typu okamžité revoluce, ale pomalé evoluce. Je to dáno jak skupinou uživatelů, kteří jsou jak v případě odborné veřejnosti (zdravotní pracovníci) tak celé společnosti, zastupování v celém věkovém spektru a jejich přístup k informačním technologiím se různí (nezávisle na věku, ale každá skupina je vyjádřena specifickým poměrem lidí s určitými znalostmi informačních technologií a jsou vysledovány rozdíly v zastoupení těchto lidí v daných věkových skupinách). Ovšem vývoj posledních let ukazuje, že postupně roste počet lidí, kteří si uvědomují přínos informačních technologií ve zdravotnictví. Dalším faktorem je i oblast zdravotnictví samotná, vyznačující se značnou nestabilitou a nechutí jakkoliv zasahovat do stávajícího způsobu spravování tohoto resortu, plus zde existují vlivné oborové organizace, s přímým podílem na chodu tohoto resortu. V tomto pohledu se ovšem EZD může vymykat, protože zde existuje silný tlak ze strany EU na projekty tohoto typu v rámci eHealth, který je veden snahou vytvořit mezinárodní prostředí pro členské země EU. (3) (4)

### 3.1 Cíl Elektronické zdravotní dokumentace

Cílem zavedení elektronické zdravotní dokumentace, ale i jiných elektronických systémů pro komunikaci, je v první řadě zrychlit a zefektivnit výměnu informace tak, aby jedinec pohybující se v tomto prostředí nebyl nositelem informace a zároveň i poskytovatelem informace. Systém (v rámci ČR, ovšem v zahraničí je situace velmi podobná) je postaven na běžné praxi, ze které vyplývá, že většina lidí využívající zdravotní péči, ať nahodile, či soustavně, nedisponuje úplnou informací o svém zdravotním stavu. Existuje sice institut praktického lékaře, který by měl být určitým centrálním bodem péče o zdraví svých pacientů, ale i ten tak velké množství informace o zdraví klienta nemá. Tyto informace jsou „rozesety“ u ostatních navazujících lékařských zařízení a jejich konsolidace je velmi obtížná a náročná. Současně je i velmi obtížné k těmto informacím včasné přistupovat, většina lékařů sice používá ambulantní softwary, ovšem ty využívá hlavně pro vyúčtování s pojišťovnou a zdravotní informace eviduje v papírové formě (tiskne z ambulantních softwarů). Tato situace se netýká velkých nemocnic, které většinou používají robustní nemocniční informační systémy, zde jsou informace uloženy v elektronické podobě. Ovšem i u těchto systémů se naráží na problém jak informaci získat a odeslat. Toto omezení je způsobené stávající legislativní úpravou, která logicky požaduje, aby u jakéhokoliv záznamu bylo řečeno, kdy byl vytvořen a podepsán vykonávajícím zdravotním pracovníkem. Krom klasického podpisu lze sice využít podpis elektronický, ale neřeší to problém s datem vytvoření, protože nynější legislativa není upravena tak, aby certifikační autorita byla zároveň i garantem tzv. časového razítka, garantujícího čas vzniku záznamu (časové razítko je možno získat, ale pouze u jedné ze současných certifikačních autorit). Jinými slovy, záznam podepsaný elektronickým podpisem nemusí nutně obsahovat informaci časového razítka.

Koncept Elektronické zdravotní dokumentace se snaží tento problém řešit. EZD vychází z předpokladu, že „majitelem“ zdravotních dat je klient (pacient), ale zároveň je i správcem a poskytovatelem těchto informací a nikoliv zdravotničtí pracovníci a zdravotnická zařízení. Zdravotnický pracovník je přispěvovatelem do systému a zároveň využívá informací v systému uložených. V praxi to znamená, že všechny informace týkající se zdravotního stavu klienta budou odesílány do systému a současně jakýkoliv oprávněný zdravotní pracovník k nim bude moci přistupovat.

### 3.2 Koncept EZD

Systém kromě přijímání, uchovávání a interpretování přijímaných zdravotních informací, by měl poskytovat další služby sloužících ke zkvalitnění zdravotní péče. Ne v pojmu lepšího léčení, to je práce lékaře, ale z pohledu bezproblémového přístupu pacienta ke svým informacím o zdravotním stavu a současně i přístup lékaře k těmto informacím.

Z pohledu koncepce je vhodné rozdělit EZD podle určitých skupin uživatelů. Tato množina skupin není úplná, subjektů, které mohou vstupovat do zdravotnictví je celá řada.

Základními EZD uživateli jsou:

- Pacient (klient)
- Zdravotní pracovník

Toto členění je případně možné rozšířit i o další subjekty vstupující do oblasti zdravotnictví, např. zdravotní pojišťovny, zdravotní zařízení apod.

Každá skupina má zcela jinou terminologii a jiné požadavky na systém EZD. Popisovaná koncepce počítá se dvěma nezávislými uživatelskými rozhraními pro každou skupinu. Snahou je maximalizovat jednoduchost ovládání a intuitivnost celého prostředí. Přístup k informacím bude odlišen na základě role, kterou daný zákazník v systému má.

Všechny služby jsou vytvořeny s těmito cíly:

- Sdílení dat se souhlasem subjektů
- Optimalizace průchodu pacienta zdravotní sítí – efektivní využití finančních prostředků veřejného zdravotního pojištění, časových možností pacienta i lékaře a kapacit zdravotnických zařízení
- Aktivní zapojení pacienta do systému zdravotní péče

Z takto koncipovaného systému plyne několik základních výhod jak pro pacienta, tak pro zdravotního pracovníka.

Z pohledu pacienta jde o následující:

- Pacient je informován o zdravotní péči, má přehled o výsledcích vyšetření i léčbě
- Zlepšuje se komunikace mezi pacientem a lékařem
- Pacient může předejít opakování některých vyšetření
- Vyhne se zbytečnému užívání léků, protože výsledný používaný lék je hodnocen na základě celkového zdravotního stavu a ne jednoho diagnostikovaného problému
- Zabránění nadužívání či interakci mezi léky
- Zrychluje se stanovení diagnózy a léčba je zahájena bez zbytečných odkladů
- Pacient má přehled o svém zdravotním stavu a o prodělaných nemocech a užívaných lécích

Z pohledu lékaře lze uvést:

- Zjednodušení komunikace mezi lékaři samotnými
- Informace jsou uvedeny lékařem a tudíž jiný zdravotní pracovník má k dispozici přesné informace o zdravotním stavu pacienta
- Informace k dispozici tehdy kdy je potřebuje
- Efektivnější léčba a zároveň nižší pravděpodobnost přelécování, či chybné léčby

- Zamezuje situacím vzniku duplicitních či nadbytečných vyšetření, preskripce již předepsaných léků apod.
- Dochází k úspoře času
- Může lépe pracovat s různými limity a regulacemi

Takto vytvořený systém umožní efektivní využívání zdravotní péče, neboť snižuje míru chyby pramenící z nedostatečné informovanosti jak zdravotních pracovníků, tak pacientů. Pacient se stává aktivní účastníkem a jako takový má zájem na co nejlepší a nejkvalitnější péči. Informace o svém zdravotním stavu má neustále přístupné a pouze on rozhoduje, komu umožní přístup, či nahlédnutí do své zdravotní dokumentace. Současně má zdravotnický pracovník k dispozici veškeré informace o zdravotním stavu svého pacienta a je mu tím pádem umožněna kvalitnější péče o pacienta. Například v případě složitějších případů může nahlížet na předešlé způsoby léčby, eliminovat již použité metody a soustředit se na konkrétní možnosti pomoci.

## 4 Standardy a komunikace v rámci EZD

### 4.1 Datové standardy

Datové standardy jako takové umožňují formální informační rozhraní určené pro komunikaci (výměnu, sdílení, vyhledávání a integraci) více zainteresovaných subjektů s tím, že každý z těchto subjektů má implementován stejný datový standard, což umožňuje snadnou a rychlou výměnu informací. V rámci zdravotnictví existuje poměrně velké množství standardů, které se zabývají různými částmi EZD. Pro potřeby této práce jsem se věnoval pouze standardům pro komunikaci a výměnu informací, tedy DS<sup>3</sup> a HL7<sup>4</sup>. Mezi další standardy patří například openEHR vyvíjené společností OCEAN Informatics, který se zaměřuje na správu a práci se zdravotními záznamy ne na výměnu dat mezi systémy (DS, HL7). Mezi další důležité standardy bych zmínil SNOMED CT<sup>5</sup>, jedná se o normalizovanou nomenklaturu, jejíž snahou je organizované systematické zpracování lékařské terminologie za pomoci informačních technologií. Hlavním přínosem SNOMED CT je, že díky způsobu jak jsou informace uloženy a normované, se jedná o konzistentní terminologii všech druhů zdravotní péče. Výsledkem je zpřesnění zaznamenaných klinických údajů, jasně definovaná struktura a velké možnosti překladu (bývá zmiňován jako standard pro sémantickou interoperabilitu). Nevýhodou je, že neexistuje český překlad, který je velmi náročný (text obsahuje kolem 310 tisíc zdravotnických pojmů).

Z pohledu zdravotnictví v rámci České Republiky je nejvhodnější použití Datového Standardu Ministerstva zdravotnictví (DS). Tento standard je v tuto chvíli podporován ve verzi 03.11.04 a 04.02.02. Hlavním důvodem pro použití tohoto komunikačního standardu je velká rozšířenost mezi subjekty, které působí na území ČR. Jedná se jak o nemocniční informační systémy, veřejné zdravotní pojišťovny, ambulantní softwary, tak i systém IZIP a ostatní.

Dalším standardem je HL7, který existuje v několika verzích. Pro účely výměny informace je v dnešní době hlavně skloňován standard HL7 verze 3 RIM (poslední). Předchozí standardy jsou stále používány, ale s jejich použitím v elektronické zdravotní dokumentaci se vzhledem k jejich zaměření, složité kontrole standardu a velmi náročné implementaci, nepočítá.

Návrh normy prEN14822, byla snaha o evropskou standardizaci ve zdravotnictví. Norma byla vytvořena CEN<sup>6</sup> za úzké spolupráce s organizací HL7. Vývoj této normy byl ukončen, je zde uvedena pouze jako informativní.

---

<sup>3</sup> DS – Datový standard MZČR

<sup>4</sup> HL7 – Health Level Seven

<sup>5</sup> SNOMED CT - Systematized Nomenclature of Medicine - Clinical Terms

<sup>6</sup> CEN - European Committee for Standardization

## 4.1.1 DASTA

Datový standard DASTA (DATový STAndard, zkratka DS a číslo verze) ve své podstatě znamená určitý datový standard pro komunikaci a vytváření dokumentu v rámci zdravotnické komunikace. Vývojem se zabývá Česká Společnost zdravotnické informatiky a vědeckých informací České lékařské společnosti Jana Evangelisty Purkyně (ČLS JEP) pod záštitou Ministerstva Zdravotnictví České Republiky.

Hlavním důvodem vzniku toho projektu MZ byl převod dokumentů pro komunikaci do obecně přístupného formátu, (dnes hlavně XML). Současně bylo nutno vytvořit i nástroje pro zavedení datového standartu pro praktické použití. Smyslem pro jeho vývoj byla automatizace přenosu dat o pacientovi a dat souvisejících mezi informačními systémy (IS) jednotlivých zdravotnických zařízení. V praxi se jedná o nemocniční IS (NIS), ambulantní IS (AIS, ISPL – informační systém praktických lékařů), laboratorní IS (LIS) apod.

DASTA definuje jednotlivé datové bloky a jejich strukturu včetně vzájemných vnořování, dokument je ohraničen elementem <dasta>datové bloky dokumentu</dasta>, na tento element (vnitřně) jsou navázány další elementy datových bloků (platné pro verzi 2 a výše). Takto strukturovaný dokument je následně předán v rámci výměny dokumentů. (3)

Poslední verzí datového standardu je verze 4, uvolněna v prosinci roku 2006 (v. DS 04.01.01) současně s NČLP 02.17.01, tato je závazná od 1.1.2007.

Vytvoření Národního číselníku laboratorních položek (NČLP) má za cíl:

- podpora další standardizace v oblasti klinických oborů s laboratorní složkou
- podpora “správné laboratorní praxe” a podpora zvyšování jakosti v laboratořích
- poskytnutí nástrojů pro rychlé zavedení Národního číselníku laboratorních položek do praxe (program ČLP)
- zajištění těsné vazby s významnými LIS (spolupráce s LIS prostřednictvím ČLP a SLP)

Dále v textu je uveden stručný přehled verzí.

### 4.1.1.1 DS 01.01

Verze byla schválena a publikována a zavedena do praxe v červenci roku 1997 (první uvolněna verze byla DS 01.00 v roce 1994). Definice je založena na pojmu datový soubor, který obsahuje jednotlivé datové bloky. Každý datový soubor je určen pro právě jednoho adresáta, tato adresa byla uvedena v datovém bloku @PM. Dalším blokem je blok @IS, který definuje odesilatele, kterých může být víc jak jeden. V tomto bloku jsou uváděna data o pacientech přenášena v datovém bloku @IP. Zápis probíhá jako prostý text, kde maximální délka řádku je 255 znaků. Ukončení bloku je pomocí znaku @.

```
@FM
IT
RLH2IVTE 11N
N
@a
P
Export pro hemokoagulaci
@AS
I
9999
@
@
251020041242
@
@IS
01.111
    001202 41049999
@a
O
4104 Hematologie - lab.
@AS
I
4104
@
@
251020041242
@
@IP
...
```

**Obr. 4-1: DS verze 01.01.**

Z obrázku (Obr. 4-1: DS verze 01.01.) je zřejmé, že tento standard byl primárně určen pro informační systémy a strojové zpracování, jeho čitelnost pro člověka je prakticky nulová. Velmi důležitým počinem provázející vznik datového standardu je vznik Národního číselníku laboratorních položek. Samotný datový standard verze 1 došel až do verze DS 01.20. a jeho podpora byla ukončena 31.12.2002.

#### **4.1.1.2 DS 02.xx.xx.**

Tato verze datového standardu byla platná od 1.6.2002 a znamenala obrovský skok kupředu. Definice a zápis byl postaven na standardu XML, což umožňovalo jak snazší implementaci, tak lepší čitelnost a celkové pochopení datového standardu. Součástí bylo uvolnění hypertextové dokumentace, kde bylo možno snadno procházet jednotlivé vazby mezi datovými bloky. Součástí definice bylo i doplnění o DTD (Data Type Definition) umožňující jednodušší validaci datového souboru a jednotlivých datových bloků. Výhodou bylo také možnost použití XSLT<sup>7</sup> pro snadné zobrazení XML v čitelné podobě.

Součástí uvolnění nové verze bylo i doplnění číselníku NČLP ve verzi 2.01.01. V rámci verze DS 02.xx.xx. došlo ke kompletní revizi datového standardu, některé datové bloky byli doplněni, upraveny, či vyjmuty.

---

<sup>7</sup> XSLT - eXtensible Stylesheet Language Transformations

#### 4.1.1.3 DS 03.xx.xx.

Verze DS 03.01.01. byla uvolněna v platnost společně s číselníkem NČLP 02.06.01 ve dne 1.1.2004 prostřednictvím věstníku MZČR, částka 9 roku 2003. Tato verze (verze 3) již vnikala za přispění konference DASTA, která slouží jako sběr a diskuse nad jednotlivými připomínkami z reálných implementací DS. Technologicky se verze 3 neliší od verze 2, ale došlo k úpravě některých datových bloků a hlavně přidání velkého množství nových. Veškeré změny jsou od této verze k dispozici na internetu (<http://ciselniky.dasta.mzcr.cz>). Od této verze také umožňuje datový standard komunikaci s Národním zdravotnickým informačním systémem (NZIS), jehož správcem a zpracovatelem dat je Ústav zdravotnických informací a statistiky ČR (ÚZIS). Touto komunikací se rozumí přidání velkého množství datových bloků (cca 60) pro možnost převést formuláře určené pro ÚZIS do plně elektronické formy. Díky této úpravě, mohou zdravotnická zařízení přímo posílat výkazy do ÚZIS čistě elektronickou formou.

Problémem souvisejícím s touto verzí bylo použití DTD. Vzhledem k velkému nárůstu datových bloků, rapidně vzrostla velikost a složitost, která činila z DTD naprosto nepřehlednou specifikaci a z toho vycházející horší aplikovatelnost v praxi. (4) (3)

#### 4.1.1.4 DS 04.xx.xx

Standard ve verzi DS 04.01.01 byl vydán společně s číselníkem NČLP 02.17.01 jako závazný od 1.1.2007. Hlavním rozdílem oproti verzi 3 je použití XML schéma (známější pod zkratkou XSD) místo DTD. V tuto chvíli je aktuální verze DS 04.02.02, v této práci se budu věnovat definici této verze.

##### 4.1.1.4.1 Struktura datového souboru

Jednotlivé datové bloky jsou definovány několika základními parametry, konkrétně jde o jméno (v rámci DS unikátní) a přesnou specifikaci přenášené informace. Definice jednotlivých datových bloků je přístupná jak pomocí XSD a DTD tak i ve formě textu (hypertext na adrese <http://ciselniky.dasta.mzcr.cz/>). Textová forma je nadřazená XSD a DTD, ale vzhledem k faktu, že není strojově zpracovatelná, tak se využívá jako dokumentace, kterou také je.

Textová podoba bloku obsahuje jeho stručný popis a použití. Dále tabulku s jeho obsahem:

- Kód – identifikátor, jméno elementu, či atributu tak jak je reprezentován v XML
- T - typ stanovující syntaktickou formu a (atribut), e (element), d (data)
- D - délka, říkající maximální počet znaků
- V – výskyt
  - ? = nepovinný, může se vyskytovat maximálně 1x ; (= 0, 1)
  - \* = nepovinný, může se vyskytovat opakovaně; (= 0 až N)
  - 1 = povinný, vyskytuje se jen 1x - v XML je default; (= 1)
  - + = povinný, vyskytuje se alespoň 1x; (= 1 až N)
- hodnota - výčet hodnot, odkaz na tabulku nebo nevyplněno



- podmínky, pokyny, poznámky – různé dodatečné informace
- změny - informace o změnách po vydání nové verze

V praxi bych ukázal základní strukturu celého dokumentu. Ten je tvořen hlavním blokem, kterým je element `<data>`. V rámci něho jsou definovány následující atributy:

- atribut `id_soubor` je povinný a slouží k identifikaci souboru v rámci společnosti a jejího programu či informačního systému.
- atribut `verze_ds` definuje verzi datového standardu a musí být shodný s verzí použitého XSD.
- atribut `verze_nclp` popisuje verzi použitého NČLP. Není-li NČLP vůbec využíván, zadává se nejnižší verze 2.00.00.
- atribut `potvrzeni` určuje úroveň detailu odpovědi.

Element `<zdroj_is>` je využit k jednoznačnému určení firmy, programu a jeho verze.

Element `<pm>` obsahuje základní informace o příjemci zasílaného souboru.

Příchozí XML dokument musí obsahovat v elementu `<vnitrni>` řetězec, určující zda je dokument určen pro specifickou bránu či systém. V tom případě se jedná o řetězec „IZIGATE“. Pokud tento není obsažen vrátí XML brána chybu.

### **Identifikace zdroje**

Pomocí elementu `<zdroj_is>` zjišťujeme používaný informační systém, či program.

Podstatné atributy jsou:

- `kod_firmy` - kód firmy, jejímž SW bylo vytvořeno XML
- `kod_prog` - kód SW, kterým bylo vytvořeno XML
- `verze_prog` - verze SW, kterým bylo vytvořeno XML

Tento element není důležitý pro validaci zprávy, ale slouží pro logování chyb, např. v případě, že určitá verze informačního systému obsahuje chyby v generování XML zpráv.

### **Identifikace lékaře**

Velmi důležitým a povinným elementem je `<garant_dat>`. Tento element identifikuje autora. Tento autor musí být zároveň tvůrcem lékařské zprávy. Element `<garant_dat>` může být obsažen v hlavním bloku `<data>` nebo se může vyskytovat v jednotlivých blocích lékařských zpráv. Pokud bude

<garant\_dat> v bloku <dasta> tak se bere jako garant celé zprávy a <garant\_dat> uvedený v jednotlivých blocích samotných lékařských zpráv vyhodnocujeme jako subgaranta. V případě, že by byl uveden jenom v bloku <dasta> a nikde jinde, pak by platil pro všechny lékařské zprávy obsažené v XML dokumentu. Pokud není garant uveden ani v jednom případě, pak je zpráva zamítnuta pro chybějící nebo nedostatečnou identifikaci zdravotního pracovníka.

Atributy elementu <garant\_dat>:

- `id_garant` - identifikace garanta, pomocí rodného čísla, či jiného povoleného identifikátoru
- `odbornost` – odbornost autora

Element <dasta> dále obsahuje jeden či více elementů <is> obsahující základní informace o odesílateli. Odesílatelem bude v případě XML brána vždy zařízení.

Atributy použité v elementu <is>:

Veškeré tyto atributy slouží k identifikaci odesílatele.

- `ico` – IČO zařízení
- `icz` – Identifikační číslo zařízení
- `icp` – identifikační číslo pracoviště
- `icl` – identifikační číslo lékaře
- `pcz` – pořadové číslo zařízení

Atribut `ico` je pro komunikaci s IZK povinný. Atribut `ico` je vždy 8-místný - pokud je IČO kratší, musí být doplněno nulami zleva. Současně je povinný i atribut `icz`, který je také 8mi místný. Atribut `pcz` je pořadové číslo pracoviště a je vždy 3místné.

Součástí elementu <is> je element <as> jenž obsahuje doplňující blok adres. Dále je součástí element <ip> určující pacienta. Těchto elementů může být v elementu <is> několik a jsou určeny jako základní blok sloužící k přenosu dat o určitém pacientovi. Co pacient to jedno <ip>, tento element obsahuje velké množství dalších atributů a elementů, z nichž některé slouží např. ke komunikaci mezi LIS (Laboratorní informační systém).

Z pohledu EZD jsou podstatné následující atributy či elementy:

`id_pac` – identifikace pacienta v IS odesílatele, jde o základní atribut pro identifikaci pacienta v EZD. K tomuto účelu může sloužit například rodné číslo (číslo pojištěnce), které je ale zároveň obsaženo v atributu `rodcis`.

- rodcis – rodné číslo
- jmeno – jméno pacienta
- prijmeni – příjmení pacienta
- <an> - anamnéza - anamnéza souhrnná neformalizovaná
- <oc> - očkování
- <le> - předepsané léky – tento element slouží k přijímání podávaných léků. Aby byla zpráva vyhodnocena jako předepsaný lék, musí být součástí elementu <ip> s jedním blokem vyšetření <z> s tím, že je uvedena před blokem <z>
- <lek> - vydané léky - léky vydané lékárnou
- <z> - ambulantní vyšetření - zpráva textová lékařská
- <z> - hospitalizace- zpráva textová lékařská
- <v> - laboratorní výsledky - výsledky vyšetření formalizované laboratorní
- <u> - urgentní informace - urgentní informace o pacientovi neformalizované
- <ku> - klinické události – jedná se o základní entitu procesu léčby pacienta. Klinickou událostí je každá událost týkající se pacienta, kterou je možné naplánovat, objednat, provést a zdokumentovat nebo vyúčtovat.

```

<?xml version="1.0" encoding="UTF-8"?>
<ds:dasta xmlns:ds="urn:cz-mzcr:ns:dasta:ds4:ds_dasta"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:dsip="urn:cz-mzcr:ns:dasta:ds4:ds_ip"
  xsi:schemaLocation="urn:cz-mzcr:ns:dasta:ds4:ds_dasta
http://ciselniky.dasta.mzcr.cz/xmlschema/ds_dasta-4.01.01.xsd urn:cz-
mzcr:ns:dasta:ds4:ds_ip http://ciselniky.dasta.mzcr.cz/xmlschema/ds_ip-4.01.02.xsd"
  id_soubor="MED_KK11115_2005-12-12T14:46:25" verze_ds="04.02.02" verze_nclp="02.23.01"
  bin_priloha="T" ur="T" typ_odesm="KK" ozn_soub="11115" dat_vb="2008-06-13T14:46:25"
  potvrzeni="P">
  <ds:zdroj_is kod_firmy="MED" kod_prog="WM" verze_prog="2.2"/>
  <ds:pm>
    <ds:as typ="I" poradi="1">
      <ds:vnitri>IZIGATE</ds:vnitri>
    </ds:as>
  </ds:pm>
  <ds:garant_dat id_garant="450124145" odbornost="801">MUDr. Jan Konečný</ds:garant_dat>
  <ds:is ico="12345678" icz="44101000" icp="44101882">
    <ds:as typ="I">
      <ds:vnitri>801</ds:vnitri>
    </ds:as>
    <dsip:ip id_pac="7601019998">
      <dsip:rodcis>7601019998</dsip:rodcis>
      <dsip:jmeno>Jmeno</dsip:jmeno>
      <dsip:prijmeni>Prijmeni</dsip:prijmeni>
      <dsip:dat_dn format="D">1976-01-01</dsip:dat_dn>
      <dsip:sex>M</dsip:sex>
      <dsip:ku>
        <dsip:ku_z duverne="N" fazespec="ZF" idku="MEDICALC.FNPL.1234567890"
typku="AMBUL">
          <dsip:dat_prov>2006-12-03T11:00:00</dsip:dat_prov>
          <dsip:dat_vydani>2006-12-05T11:00:00</dsip:dat_vydani>
          <dsip:p_pracoviste icz="44101000" icp="44101882" odb="101">

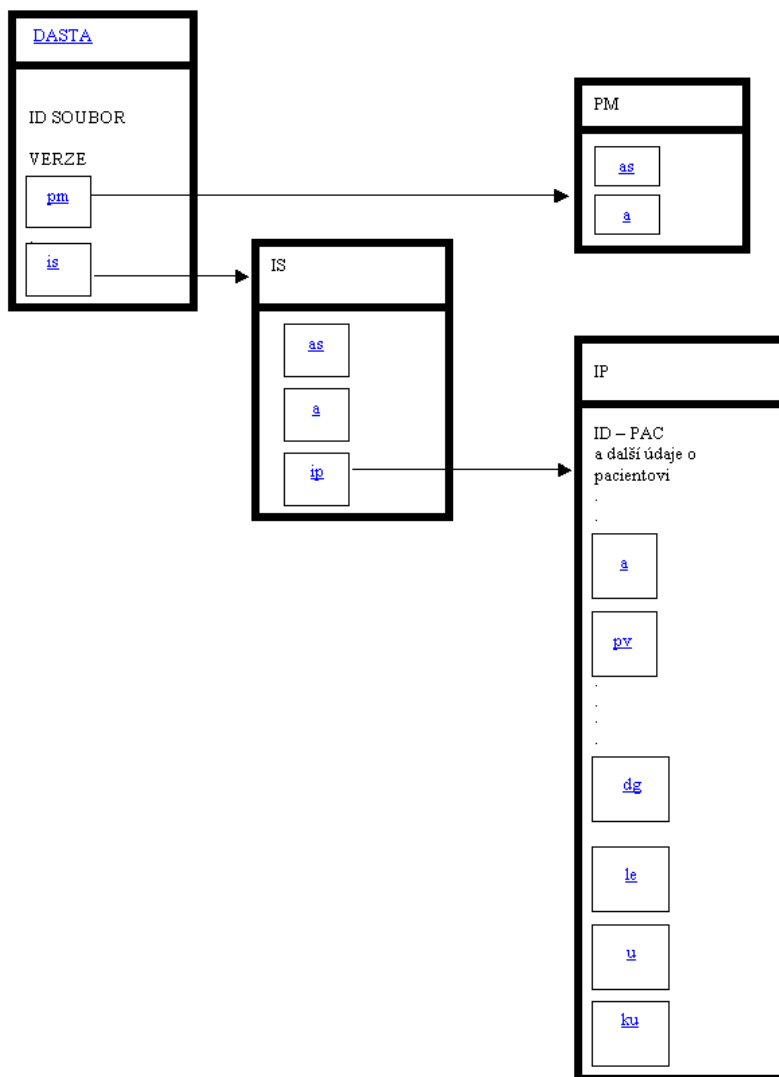
```

```

<dsip:nazev>Interní ambulance</dsip:nazev>
</dsip:p_pracoviste>
<dsip:text>
  <dsip:ptext xml:space="preserve">Nález normální, za 3 týdny doporučena
kontrola u praktického lékaře.</dsip:ptext>
</dsip:text>
<dsip:dg_vys typ_dg="P" ind_oprav_sd="N">
  <dsip:diag poradi="1">I158</dsip:diag>
</dsip:dg_vys>
<dsip:auzku indikace="I" typpol_vz="V" diag="I158">
  <dsip:dat_du typ="A">2006-12-03T11:00:00</dsip:dat_du>
  <dsip:vykon kod_vykonu="11023" pocet="1" uctujici="0"/>
  <dsip:p_pracoviste icz="44101000" icp="44101882" odb="101">
    <dsip:nazev>Interní ambulance</dsip:nazev>
  </dsip:p_pracoviste>
</dsip:auzku>
</dsip:ku_z>
</dsip:ku>
</dsip:ip>
</ds:is>
</ds:dasta>

```

**Příklad 4-1: příklad datového souboru DS 04.02.02 - klinická událost**



**Obr. 4-2: DASTA - blokové schéma**

#### 4.1.1.4.2 Číselníky NČLP, DS, ÚZIS

Součástí datového standardu jsou i jednotlivé číselníky NČLP, DS, ÚZIS. Jedná se o unikátní soubor dat, jehož smyslem je jednoznačná identifikace termínů používaných ve zdravotnictví a zabránit tak neporozumění lokálně používaným synonymům pro daný termín. Jak již vyplývá z názvu, jsou tyto číselníky rozděleny do tří skupin - NČLP, DS a ÚZIS.

V praxi je NČLP (i LČLP<sup>8</sup>) nejvíce využíván Datovým standardem MZ ČR (DS), laboratorními informačními systémy (LIS), některými nemocničními informačními systémy (NIS) a předpokládá se využívání propracovanějšími informačními systémy praktických lékařů (ISPL). NČLP je dále využíván při tvorbě standardů laboratorních a klinických oborů, při tvorbě encyklopedií laboratorních oborů, textů repertorií pro výuku atd.

Základní strukturou, která je obsahem jakéhokoliv číselníku, je položka klíč a xx (název, kde xx je číslo délky). Klíč je unikátní, slouží jako identifikátor dané položky. Všechny tyto číselníky jsou k dispozici jak na CD tak formou webových služeb. (3)

##### Číselník NCMPATML

Číselník	NCMPATML
Název	Číselník antimikrobiálních látek
Zdroj	Číselníky NČLP
Aktualizace	1.1.2009 22:35:43
Klíč(e)	KLIC
Počet vět	104
Sada	200910
Sada změna	200910
Verze NČLP	02.23.01
Verze DS	04.02.02
Platnost od	1.1.2009
Platnost do	31.3.2009

KLIC	N32	N55	PORADI
AMC	amoxicilin klavulanát	amoxicilin klavulanát	001000
AMF	amfotericin B	amfotericin B	002000
AMI	amikacin	amikacin	003000
AMP	ampicilin	ampicilin	004000
AMS	ampicilin sulbaktam	ampicilin sulbaktam	005000
AMX	amoxicilin	amoxicilin	006000
AZI	azitromycin	azitromycin	007000
AZL	azlocilin	azlocilin	008000
AZT	aztreonam	aztreonam	009000
BAC	bacitracin	bacitracin	010000

##### Příklad 4-2: příklad NČLP číselníku

<sup>8</sup> LČLP – lokální číselník laboratorních položek

## 4.1.2 HL7

HL7, neboli Health Level Seven, je dobrovolná nezisková organizace řízená správním radou. Struktura organizace je rozdělena na stálé správní výbory, které se zabývají hlavně organizační a propagační činností a tzv. pracovní skupiny, které se zabývají přímo obsahem norem a vytvářením specifikace samotného standardu. Tato organizace se podílí se na rozvoji mezinárodních zdravotních norem na rozdíl od DASTA, která pokrývá hlavně problematiku českého zdravotnictví. HL7 byla založena roku 1987 za účelem vypracování norem pro nemocniční informační systémy pod akreditací organizace ANSI<sup>9</sup>, tuto akreditaci získala v roce 1994. Postupem času vývoj byl HL7 rozšířen o řadu standardů vzniklých pod SDO<sup>10</sup>, které již nejsou akreditované ANSI, nicméně v současnosti je HL7 přijata jako standard ISO<sup>11</sup>. První vzájemně zveřejněnou normou je ISO/HL7 21731:2006, neboli HL7 verze 3 - RIM<sup>12</sup>. Název HL7 vychází z odkazu na referenční model ISO/OSI, kde sedmá úroveň je aplikační vrstva, což je vrstva kterou HL7 postihuje pro použití v rámci zdravotní péče. Tato úroveň popisuje definici vyměňovaných dat, časování výměny, podporuje funkce pro bezpečnostní kontroly, kontroly dostupnosti a zejména strukturování dat, všechny ostatní vrstvy používá pouze jako nástroje. (7) (6). Organizace HL7 úzce spolupracuje s ostatními tvůrci standardů, jako například ASTM<sup>13</sup>, ACR<sup>14</sup>, X12N<sup>15</sup>, IEEE<sup>16</sup>.

V textu se soustředím na základní popis standardu HL7, kdo a jakým způsobem ho může používat, základní zobrazení struktury dokumentu v HL7, skladbu artefaktů dokumentu apod.

Stejně jako název organizace je nazván i datový standard, přičemž se jedná se o ve světě nejpoužívanější standard sloužící k výměně, integraci, sdílení a vyhledávání elektronických zdravotních informací. Existuje několik verzí, mezi nejpoužívanější patří HL7 v2.x, v3.0, HL7 RIM.

### 4.1.2.1 HL7 v2.x

Verze HL7 v2.x se vyznačuje vývojem „zdola nahoru“, který řešil vždy jednotlivé potřeby za použití ad-hoc metodologie. Byť byla tato verze velmi úspěšná a široce implementována, setkáme se s ní kupříkladu u různých diagnostických přístrojů, není plně vyhovující právě díky způsobu vývoje. Ten způsobil, že verze 2 neposkytuje konzistentní pohled na přenášená data a ani vztah těchto dat k jiným datům, jinými slovy mezi jednotlivými elementy není žádný referenční vztah tak jak ho známe např. u objektového přístupu. Úspěch této verze lze přičíst její flexibilitě, neboť obsahuje mnoho volitelných datových

---

<sup>9</sup> ANSI - Americký Národní institut pro normalizaci

<sup>10</sup> SDO - Standard Developing Organization

<sup>11</sup> ISO - Mezinárodní organizace pro normalizaci

<sup>12</sup> RIM - Referenční Informační Model

<sup>13</sup> ASTM - American Society for Testing and Materials

<sup>14</sup> ACR - American College of Radiology

<sup>15</sup> X12N - Accredited Standards Committee

<sup>16</sup> IEEE - Institute of Electronics and Electrical Engineers

elementů a segmentů, které umožňují adaptaci standardu na téměř jakékoliv prostředí. Ovšem současně tato volnost způsobuje, že je nemožné provádět spolehlivé testy na shodu se standardem (jako např. v případě validace XML dokumentu vůči definovanému XSD<sup>17</sup> schématu) pro jakoukoliv implementaci a je nutné, aby implementaci předcházela rozsáhlá analýza dílčích rozhraní a bylo zajištěno, že obě (a více) strany budou používat stejné volitelné segmenty a elementy. Z těchto důvodů byl vytvořen ve své podstatě nový standard HL7 v3.

#### 4.1.2.2 HL7 v3

V případě verze 3 byl zvolen odlišný přístup k vývoji standardu oproti verzi 2. Verze 3 řeší problémy popsané v 4.1.2.1 HL7 v2.x tím, že používá dobře definovanou objektově orientovanou metodologii založenou na referenčním informačním (datovém) modelu (RIM), který vyjadřuje obsah dat v kontextu klinických a administrativních informací. Tím zajišťuje explicitní znázornění sémantického a slovníkového (lexikálního) spojení, které existuje mezi informacemi přenášenými za pomoci standardu HL7 v3. Tento standard nabízí jednoznačnost, jednodušší testování a tím i markantní snížení nákladů nutných na jeho implementaci. Převážně díky tomu, že je nutno důsledně používat přesné analytické techniky a postupy pro tvorbu zpráv a zařazení více spouštěcích událostí, kde formát zpráv umožňuje minimální volitelnost. (6)

Mezi nové možnosti verze 3 patří:

- Přístup „shora dolů“ při tvorbě zpráv s důrazem na znovupoužitelnost ve více kontextech a sémantickou interoperabilitu (kombinace se SNOMED CT)
- Reprezentaci komplexních vztahů
- Formalismus pro podporu slovníků
- Podporu integrace ve velkém měřítku
- Jednotnou množinu modelů
- Řešení opětovného použití a interoperabilitu mezi kontexty více domén<sup>18</sup>
- Rozšířenou působnost na epidemiologii, veterinární medicínu, klinickou genomiku, bezpečnost atd.

#### 4.1.2.3 Struktura standardu HL7 v3

HL7 v3 se skládá z řady dokumentací, z nichž některé se věnují samotné specifikaci, zatímco jiné se soustředí na informace, které jsou důležité pro vývoj a přenos zpráv za pomoci HL7. Mezi základní komponenty HL7 patří informační modely, statická struktura, slovníky, datové typy, CMETs<sup>19</sup> a ITS<sup>20</sup>. (7)

---

<sup>17</sup> XSD – XML schema definition

<sup>18</sup> Doména – v tomto kontextu se rozumí určitá oblast ve zdravotnictví, ne např. doména internetová

<sup>19</sup> CMETs – Common Message Element types – Společné typy prvku zprávy

<sup>20</sup> ITS – Implementation Technology Specification - Specifikace implementační technologie

#### 4.1.2.3.1 Informační modely

Informační model je soubor dokumentu popisující jednotlivé typy informačních modelů používaných ve standardu HL7 v3, včetně Referenčního informačního modelu (RIM). Jde vlastně o strukturovanou specifikaci informací, která popisuje třídy požadovaných informací a vlastnosti těchto tříd, včetně atributů, vztahů, omezení a vztahů. HL7 definuje různé typy informačních modelů, které umožňují reprezentovat různé oblasti zájmu. Tyto modely se skládají z následujících komponent:

- Třídy, jejich atributy a vztahy mezi třídami
- Datové typy pro všechny atributy a slovníkové domény pro kódované atributy
- Stavové modely pro některé třídy

Všechny informační modely HL7 jsou založeny na modelovacím jazyce UML a lze je tedy možno interpretovat za pomoci grafických prvků jazyka UML. Proces modelování informací v HL7 v3 využívá tři spolu souvisejících typů informačních modelů, přičemž každý z použitých typů používá stejnou notaci a má stejnou základní strukturu. Odlišují se svým informačním obsahem, zaměřením a použitím. Jde o Referenční informační model (RIM), Doménový informační model (D-MIM<sup>21</sup>) a Zpřesněný informační model zpráv (R-MIM<sup>22</sup>).

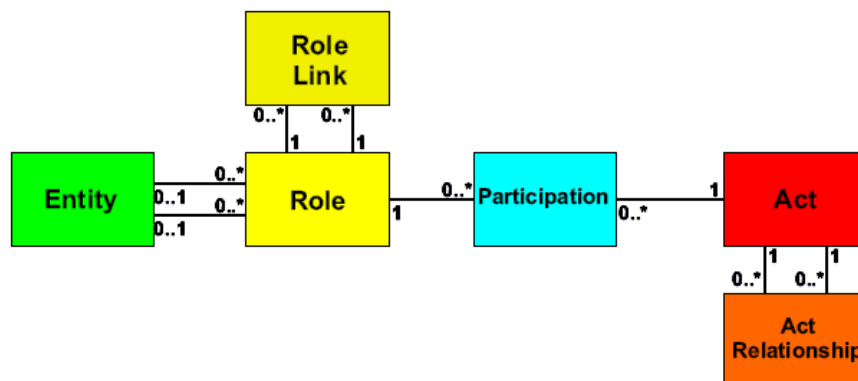
**Referenční informační model** – RIM se používá pro vyjádření informačního obsahu. Jedná se tedy o informační model, který pokrývá standard HL7 jako celek. RIM je soudržný sdílený informační model, který je zdrojem datového obsahu všech zpráv vytvořených v HL7 v3 a nese v sobě základní definice tříd, jejich atributů a vzájemných relací mezi třídami. Snaha organizace HL7 je, aby RIM byl univerzálním datovým modelem, pomocí něhož lze modelovat většinu situací ve zdravotnictví. RIM není uzavřený a neustále dochází k jeho postupnému vývoji, jakákoliv nová verze však podléhá přísnému připomínkování a schvalování ze strany HL7 Technical Committee, aby model neobsahoval zbytečné věci. V současné době obsahuje RIM více jak 70 tříd včetně vzájemných vazeb a dědičností. Každá třída má své atributy, které mají přesně určený datový typ, použití a četnost výskytu (0:N, 0:1, 1:N apod.). Na obrázku níže je vidět základní kostra RIMu pro přenášené datové struktury.

---

<sup>21</sup> D-MIM – Domain Message Information Model

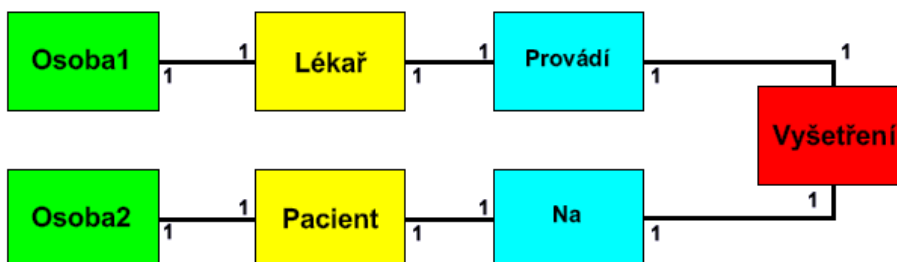
<sup>22</sup> R-MIM – Refined Message Information Model





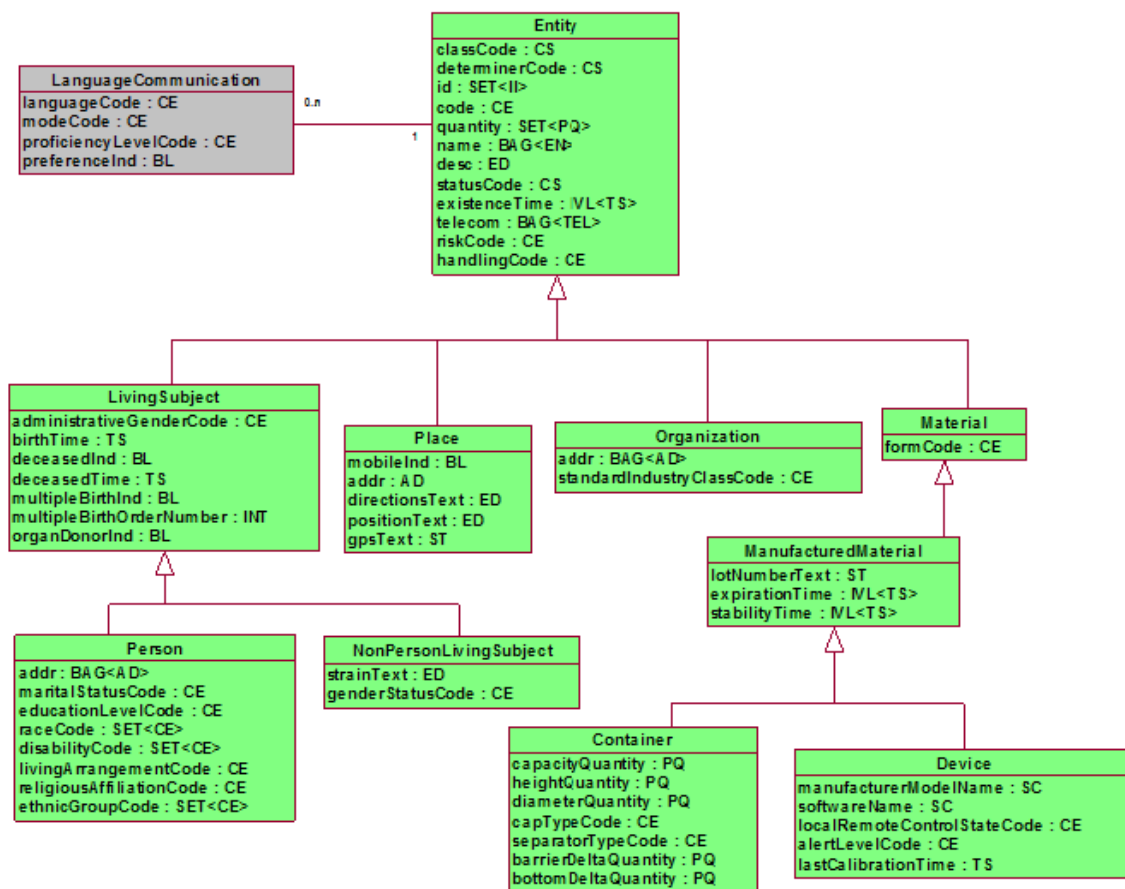
Obr. 4-3: kostra Referenčního Informačního Modelu (zdroj HL7 NPO)

Jak je možno vyzpozorovat z Obr. 4-3: kostra Referenčního Informačního Modelu (zdroj HL7 NPO) hlavní struktura RIM je tvořena několika základními objekty. Těmi jsou Entita, Role, Akce a dále spojení mezi Akcí a Rolí, nazvané Participace.



Obr. 4-4: příklad reprezentace zdravotního výkonu v RIM HL7

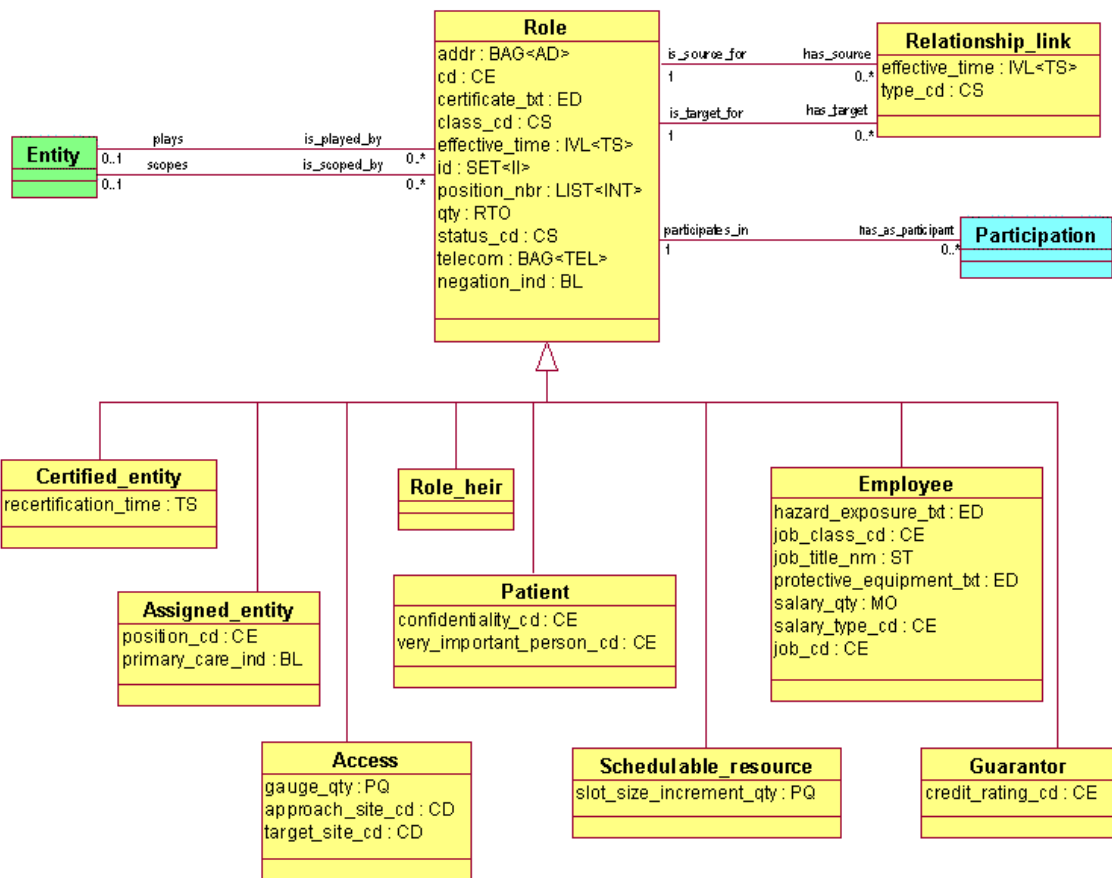
**Entita** – jde o objekt, který označuje určitý neživý, živý předmět, místo, organizaci apod. Tyto Entity nesou specifikace živých objektů (člověk, zvíře), organizací, léčebných látek, materiálů, míst a věcí (zařízení a nástroje). Z podstaty tohoto objektu je zřejmé, že nelze určit, jakou roli bude entita mít či jaké akce se bude účastnit. Například entita označená jako „Albert“ vedená jako „Person -> LivingSubject -> Entity“ může v jednom případě mít roli „lékař“ a vykonávat akci „operace“ a jiném roli „pacient“ a akci „operován“. Toto byl jenom příklad, Entita může být např. „obvaz“, „ibalgín“, „Nemocnice Na Františku“, „kočka“, „člověk“ apod.



Obr. 4-5: hierarchická struktura třídy Entita (zdroj HL7 NPO)

**Role** – významem tohoto objektu je definice úlohy Entity. Jedná se o určitou klasifikaci chování (u živých entit) a způsob použití (u neživých Entit). Samotná role je určena jednou Entitou označovanou jako „player“ (hráč). Jde o aktivní entitu a druhou entitou „scoper“ (pozorovatel), zde se jedná o pasivní Entitu. Názorně lze předvést na příkladu, kdy *player* je rolí pro pacienta a *scoper* je zdravotnické zařízení, ve které je daný člověk pacientem.

Principem Role je reprezentace a definice kompetencí dané Entity podle nároků dané Entity. Entity se podle definovaných Rolí stávají předmětem či účastníkem (Participace) určité Akce. Typickým příkladem Rolí je pacient, zaměstnanec aj. pro neživé předměty bývá jejich rolí např. definice jejich účelu, vstup u skalpelu (pro předměty které pronikají do těla za účelem operace, odběr tkání apod.). Dále je v rámci Rolí možno specifikovat vazbu mezi jednotlivými rolemi a to za pomoci RoleLink (na obrázku Obr. 4-6: hierarchická struktura třídy Role (zdroj HL7 NPO) je označen jako „Relationship\_link“), která slouží k definování vazeb hierarchických struktur mezi jednotlivými rolemi (využití např. pro definici organizační struktury v organizaci - ředitel, primář, lékař).



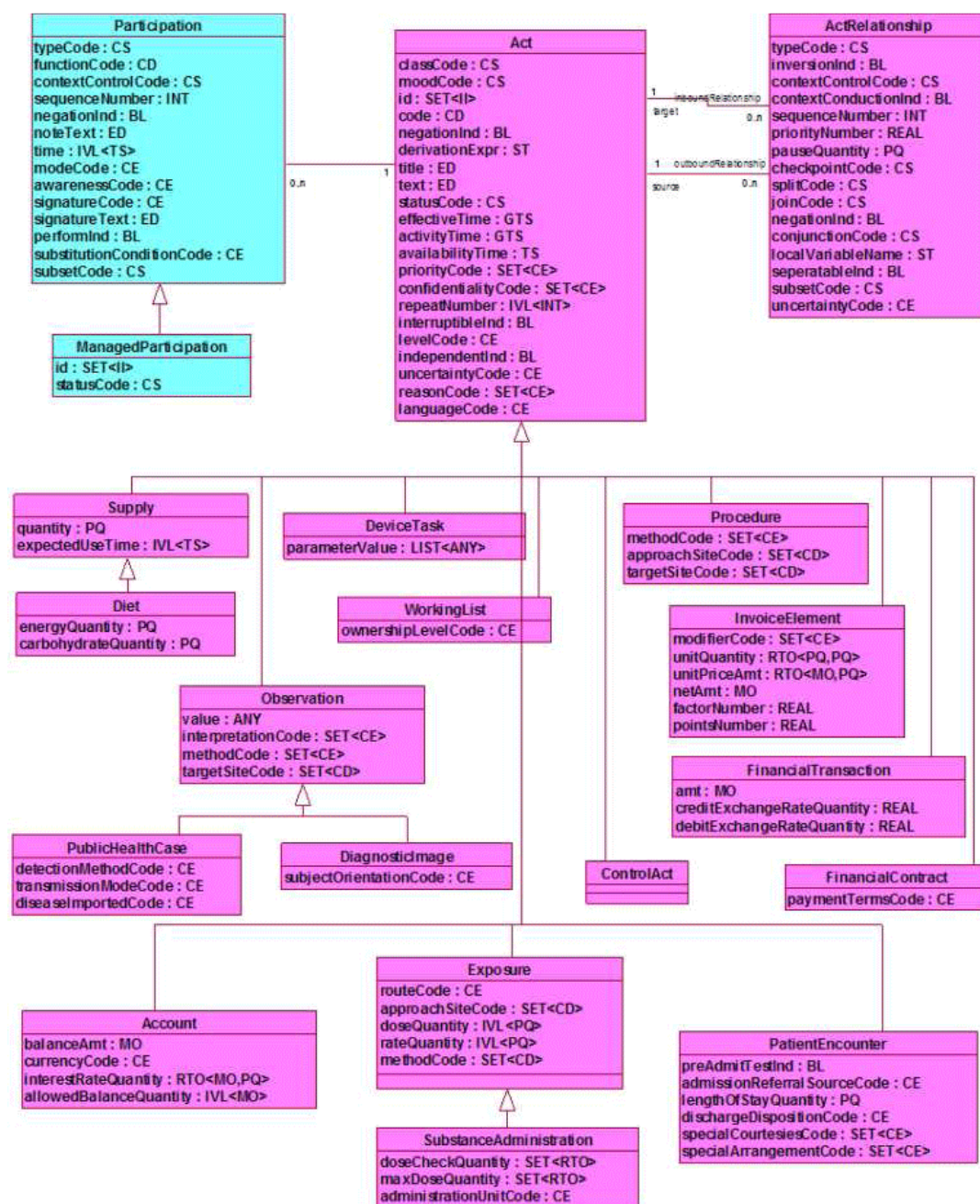
Obr. 4-6: hierarchická struktura třídy Role (zdroj HL7 NPO)

**Akce (Act)** – pokud hovoříme o události v HL7, máme na mysli objekt s ní spojený respektive záznam určité události, která se stala, děje se, nebo se v budoucnosti stane. V záběru zdravotnictví se může jednat o lékařské vyšetření (s diagnózou), klinické pozorování, léčebné úkony apod. Tyto události (akty) mají atribut `moodCode`, který stanovuje účel dané události, tedy zda se jedná o vyjádření explicitní situace, příkazu či možnosti. Dalším atributem je nepovinný atribut `statusCode` říkající nám v jakém aktuálním stavu se událost nachází (aktivní, zrušený, obsoletní...). Pokud je akt vyvolán jiným aktem, nebo je následován dalším aktem, sleduje se tato událost přes vazbu `V ActRelationship`.

**Participace** – jde o vazbu, která spojuje Akci a Roli. Velmi důležitým atributem je `typeCode`, který říká jakým způsobem je Entita s definovanou Rolí zapojena do Akce (Aktu). Z pohledu medicínského prostředí se může jednat např. o:

- Subjekt (předmět aktu) – pacienti, přístroje, nástroje
- Autor, svědek, asistent apod.
- Místo akce
- Adresát
- Vykonávající – lékař praktik, lékař ortoped, medicí apod.

Při pohledu na RIM a jeho pojetí Role, Participace a dalších souvisejících tříd (na obrázku: Obr. 4-3: kostra Referenčního Informačního Modelu (zdroj HL7 NPO)), je vidět, že Role společně s Participací leží uprostřed mezi Entitou a Akcí. Entita zahrnuje fyzické záležitosti, organizace a místa. Třída Act reprezentuje záznam o záměrné akci v prostředí zdravotnictví. V tomto spojení Role reprezentuje kompetence Entity podle jejího typu. Entity s určitou Rolí mohou participovat v Akcích ovšem mnoha způsoby, protože Participace popisuje konkrétní způsoby, jak se Entita chová v Akci. Aby byl rozdíl mezi významem Role a Participace co nejzřejmější lze uvést, že: „Participace představuje vykonání, zatímco Role představuje způsobnost“.



Obr. 4-7: hierarchická struktura třídy Participace a Akce (zdroj HL7 NPO)

Jako takový je tento model RIM záměrně abstraktní, díky čemuž by měl umožňovat reprezentovat většinu velmi variabilních informací sdílených v rámci zdravotního systému. (8) (6) (9)

**Doménový informační model – D-MIM** - je přesnější podmnožina objektů RIM. Tato podmnožina obsahuje množinu tříd, atributů a vztahů, které mohou být využity při tvorbě zpráv pro určitou doménu respektive pro určitou oblast zdravotnictví např. elektronické předepisování léků. D-MIM je určitý bazální rámec, na kterém jsou vystavěny všechny R-MIM v rámci domény.

**Zpřesněný informační model – R-MIM** – je podmnožina D-MIM použitá pro vytváření informačního obsahu zprávy, či množiny zpráv s anotacemi a zpřesněními specifickými pro tyto zprávy. V rámci D-MIM pro danou doménu může zpřesněný informační model využívat odvozené třídy z vybraných tříd s aliasy, specifické pro pohled na danou doménu. R-MIM reprezentuje informační obsah pro jednu nebo více abstraktních struktur zpráv (HMDs<sup>23</sup>). (7)

#### 4.1.2.3.2 Statická struktura a atributy

Statická struktura vychází z celkového objektového návrhu a je tvořena třídami, jejich instancemi, tedy objekty, vztahy mezi třídami (generalizace, asociace). Obsah přenášené informace je nesen atributy tříd, které rozdělujeme do tří typů, identifikační, klasifikační a stavové atributy.

Identifikační atributy jsou použity pro identifikaci instance třídy a hodnota atributu musí být vždy unikátní mezi všemi instancemi dané třídy (někdy je pro identifikaci instance potřeba více než jeden atribut). Jde o identitu statickou, hodnota atributu se tedy nemění, nejčastěji bývá název atributu ve tvaru „id“ a typ „set of instance identifier“ (SETII).

Klasifikační atributy jsou hlavní součástí tříd formulujících strukturu RIM (Entita, Role, Akce). Každá z těchto tříd má společný atribut „classCode“, který je ze slovníku domény náležející dané třídě a určuje přesně její typ. Tato vlastnost umožňuje informačnímu modelu relativně vysokou pružnost. Slovníkové domény pro klasifikační atributy obsahují záznam pro každou specializaci páteřní třídy. Kupříkladu zápis ve třídě Entita může vypadat následujícím způsobem: `Entity.classCode` má elementy „subjekt“, „organizace“, „místo“. Slovníková doména ovšem může také obsahovat záznamy, které nejsou explicitně vyjádřeny jako třídy modelu.

Stavové atributy jsou používány v předmětových třídách. Jejich obsahem jsou hodnoty, které vyjadřují současný stav třídy. Předmětová třída smí mít pouze jeden stavový atribut, jehož datový typ je „set of code value“, který umožňuje specifikovat více stavových příznaků. Jednotlivé stavové atributy se jménem „status\_cd“ jsou asociovány se slovníkovou doménou, které koresponduje se stavovým stroje definovaným pro předmětovou třídu. Např.

---

<sup>23</sup> HMDs – Hierarchal Message Definitions – Hierarchické popisy zpráv

Act.Status\_cd může mít hodnotu „dokončený“, což znamená, že daná akce byla úspěšně ukončena. Dalšími hodnotami jsou aktivní, pozastavený, zrušený, nezdařený.

Omezení slouží k zúžení množiny hodnot, kterých mohou atributy nabývat. Typem omezení, může být kupříkladu omezení na slovníkovou hodnotu, které říká, že daný atribut musí kód LOINC<sup>24</sup>, nebo omezení na rozsah hodnot (0-1 apod.). Lze ho definovat na úrovni RIM, D-MIM, R-MIM nebo HMD. Omezení na vyšší úrovni modelu platí pro všechny odvozené modely a nemůže být na nižší úrovni odstraněno. Pokud tedy nadefinuji omezení na úrovni D-MIM, mohu jej změnit v RIM, ale ne v R-MIM. (4) (9)

#### 4.1.2.3.3 Slovníková doména

Slovníková doména v HL7 v3 RIM je množina pojmů, které mohou být platnými hodnotami v instanci kódované položky nebo atributu, nejedná se ovšem o množinu definovaných slov nebo kódů. V různých implementacích rozhraní může být stejný pojem reprezentován různými kódovacími systémy. Proto má každý pojem ve slovníkové doméně vztah "jeden k mnoha" ke kódům, které mohou být použity jako reprezentace pojmu v instanci zprávy.

Každý kódovaný atribut v RIMu je asociován právě s jednou slovníkovou doménou. Některé slovníkové domény jsou asociovány s více než jedním atributem RIMu. Slovníková doména může být tabulka definovaná v HL7 a uložená v repository HL7, externí kódovací schéma uznávané HL7 (např. LOINC, SNOMED) nebo jejich kombinace. Kódované položky obsahují dva údaje vztahující se ke slovníku: jméno slovníkové domény a kvalifikátor rozšiřitelnosti, který má dvě možné hodnoty: CNE (coded no extensions), a CWE (coded with extensions). Pro hodnotu CWE mohou být jako hodnota kódovaného atributu ve zprávě použity lokální pojmy nebo volný text, pokud požadovaný pojem není ve standardní slovníkové doméně.

#### 4.1.2.4 **Srovnání HL7 a DASTA**

Datový standard (DS) je velmi propracovaný nástroj pro výměnu informací, snaží se pokrýt většinu druhů informací vyměňovaných v rámci zdravotnictví a z mého pohledu se mu to daří. Ze zkušenosti s verzemi 2.xx.xx, 3.xx.xx a 4.xx.xx vím, že implementovatelnost tohoto standardu je relativně snadná, k čemuž přispívá velkou měrou velmi dobrá dokumentace a uvedené příklady. Dalším obrovsky významným počinem jsou číselníky, které mají za úkol jednoznačnou unikátní identifikaci položek a současně obsahují základní definice a popisy. Výsledkem je zdravotní informace, která nese jednoznačnou strojově zpracovatelnou informaci, toto se ovšem týká hlavně laboratorních vyšetření a diagnóz. V případě textových vyjádření (anamnéza, závěr vyšetření) je přenášen pouze nestrukturovaný text. Jako takový má datový standard pevnou strukturu, veškeré bloky jsou jednoznačně definované. Tato vlastnost má za následek sice jednodušší implementaci, ale pokud v datovém standardu

---

<sup>24</sup> LONIC - Logical Observation Identifiers Names and Codes

není určitý druh přenášené informace přesně nadefinován nelze takovou informaci přenášet.

Z pohledu zdravotnictví jako celku datový standard řeší pouze úroveň samotné výměny informací, kdy například praktický lékař odešle do laboratoře žádanku o rozbor krve a laboratoř na základě této zprávy odešle výsledek. Nejde o standard, který by popisoval historii a stav situace, hierarchickou strukturu zaměstnanců, ekonomické informace o zákrocích, práva na určité zařízení, výkony apod.

Naopak, HL7 v3 RIM z principu jejího návrhu, kdy staví na univerzálním referenčním informačním modelu, umožňuje popsat téměř jakékoliv informace vnikající v prostředí zdravotní péče a jejich převod do datových struktur. Forma komunikace v HL7 je chápána jako určitý otevřený systém, ve kterém je přijat požadavek, na základě kterého postupně přicházejí odpovědi z různých částí systému. Principem této komunikace je vyslání požadavku a čekání na data, tento standard jako celek je navržen na formu komunikace otázka - odpověď (request - response) s možností přenést v podstatě jakékoliv data. Umožňuje například vznést dotaz, jestli v daném zdravotním zařízení je k dispozici magnetická rezonance v určitý termín, pro jistý druh vyšetření.

V porovnání HL7 oproti DS se HL7 zdá být všestrannější a univerzálnější, tato výhoda je ovšem za cenu enormních nároků na implementaci a také faktem, že jeho kódový slovník není v tuto chvíli definován. Naproti tomu sice DS neumožňuje pohled na zdravotnictví v tak podrobném pohledu a možnost modelování, ale jeho implementace je několikanásobně snazší a přítomnost rozsáhlých číselníků je velmi cenný nástroj z pohledu jasné interpretace přenesených informací. Výstupem lze těžko říci, který z popsaných datových standardů považovat za více vyhovující. HL7 je komplexní, nabízí kompletní komunikační metodiky pro výměnu, získávání a interpretaci zdravotních záznamů, počítá s použitím SNOMED CT, pracuje nejenom se zdravotním záznamem, ale s celou historií vyšetření, ovšem za cenu, kdy implementace je enormně náročná. To je také hlavní kritikou tohoto standardu, kdy mu vytýkají, že po investicích milionů dolarů do implementace v informačních systémech je nutno dělat vlastní úpravy aby bylo možno systém vůbec používat. (10)

## 4.2 Webové služby

Častým problémem současných systému je nedostatečná možnost jejich vzájemné integrace. Pro potřeby EZD je velmi důležité aby systémy fungující ve zdravotnickém segmentu byly schopny spolu komunikovat, vyměňovat si data a zpětně získávat odpovědi. Tento stav je dán časem a účelem vzniku těchto aplikací a systému, kdy interoperabilita nebyla na pořadu dne. Tyto překážky jsou v prvé řadě technologického charakteru a dají se řešit. Mezi technologické překážky lze vyjmenovat následující:

- Různé transportní protokoly
- Odlišné nároky na bezpečnost (šifrování, podpis)
- Nejednotný formát výměny dat (csv, xls, binární...)

- Různá kódování dat (latin, win-\*, iso-\*, ...)
- Způsoby komunikace (synchronní, asynchronní)
- Odlišná programová prostředí (C++, JAVA, .NET, PHP...)
- a další.

Mezi další překážku lze zařadit i složitou případně nedostatečnou či zcela chybějící dokumentaci nástrojů komunikace.

Lze říci že hlavním problémem je chybějící jednotný standard komunikace mezi systémy. Jedním z řešení a z mého pohledu správným je využití Webových Služeb (Web services), dále v textu budu používat i zkratku WS.

Webové služby (WS) lze chápat jako určitou množinu standardů, jejímž cílem je vytvořit jednotné podmínky komunikace mezi navzájem heterogenními systémy. Standardy vyvíjí konsorcium W3C a OASIS. Některé společnosti jako ORACLE, IBM, HP, Microsoft v rámci vývoje jejich integračních řešení taktéž vyvíjejí standardy spadající pod skupinu WS, ale spíše lze říci bohužel, protože dochází k modifikaci stávajících standardů a výsledkem je opět nejednotný standard. K tomuto ale nedochází příliš často a v poslední době se i tyto velké společnosti snaží dodržovat jednotnou specifikaci. Jejich motivací, ale není respektování schválených standardů, ale snaha aby se drahá řešení těchto společností vůbec domluvila mezi sebou.

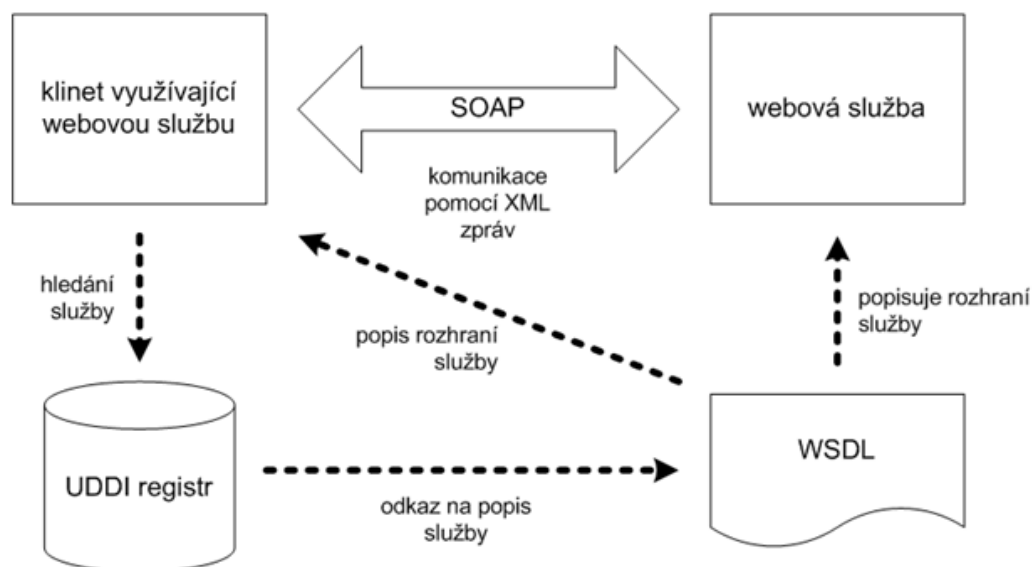
Webové služby, jak již bylo napsáno, jsou souborem jednotlivých standardů, které jako celek zajišťují jednotný koncept komunikace mezi systémy. Mezi nejdůležitější bych zmínil následující.

Standardy používané v rámci WS:

- HTTP – jednotný transportní protokol
- SOAP – jednotný komunikační protokol
- WSDL – samopopisující jazyk WS (dokumentace)
- XML(XSD) – jednotný datový formát zpráv
- UTF-8 – jednotné kódování (lze použít i jiné)
- WS - \* - standardy pro bezpečnost, routing, způsoby komunikace apod.

Ideální případ jednoduché komunikace je znázorněn na schématu Obr. 4-8: schéma komunikace pomocí WS.





Obr. 4-8: schéma komunikace pomocí WS

## 4.2.1 Stručný popis jednotlivých specifikací

### 4.2.1.1 Transportní protokol

V terminologii WS není pojem transportní protokol shodný s protokoly obsažené v rámci tzv. transportní vrstvy TCP/IP modelu (tj. TCP, UDP...). Ve smyslu transportního protokolu pro potřeby WS se většinou rozumí protokoly aplikační vrstvy TCP/IP modelu (HTTP, FTP, SMTP...). Ve spojení s WS se nejčastěji hovoří o protokolu HTTP. Dle definice W3C<sup>25</sup> (a reálné praxe) lze ale použít i jiné protokoly (SMTP, FTP...).

Ve valné většině se, ovšem používá protokol HTTP hlavně kvůli svému masivnímu rozšíření (síť Internet, proxy) a velké dostupnosti softwaru pro jeho provoz (web server). S využitím tohoto protokolu jsou spojeny ovšem i určité nevýhody, tento protokol je bezstavový, proto ve chvíli kde je potřeba udržovat jeden určitý stav pro více požadavků v čase, je nutno toto obcházet použitím session (cookies). Dále nebyl navržen pro transport a jako takový nemá prakticky žádné zabezpečení. Ovšem všechny tyto nevýhody lze řešit, zabezpečení přidáním SSL vrstvy, pro transport je už beztak dlouho využíváno a jeho rozšíření a snadné použití vyváží případné problémy. (11)

### 4.2.1.2 Komunikační protokol SOAP

SOAP<sup>26</sup> je specifikace protokolu sloužící pro výměnu strukturovaných informací v implementaci WS v rámci počítačových sítí. Je definován jazykem XML a současně přenášená informace je strukturována v jazyku XML. SOAP vytváří ve své podstatě základní vrstvu pro komunikaci mezi jednotlivými WS.

<sup>25</sup> W3C – World Wide Web Consortium

<sup>26</sup> SOAP – Simple Object Access Protocol

SOAP je založen na standardu XML, skládá se z elementů `envelope`, `header` a `body`. Element `envelope` je kořenový element, který zapouzdřuje zprávu jako celek. Element `header` je tvořen obecnými údaji o zprávě jako je typ autentizace apod. Může zde být uvedeno, prakticky cokoliv. Tento element je nepovinný. Hlavní část SOAP zprávy je tvořena elementem `body`, tento element je povinný a obsahuje název volané metody a parametry metody.

Velkou výhodou SOAP protokolu je jeho dostatečná obecnost, základní definice jsme schopni velmi jednoduše rozšířit o další volané metody. Velmi důležitou vlastností je jeho nezávislost na platformě či použitém programovacím jazyku. Krom výhod má SOAP i několik nepříjemných nevýhod, jednou z nejpodstatnějších je problematické přidání binárního souboru do zprávy, lze řešit buď přidáním do SOAP `body` nebo jako `multipart` pomocí MIME či MTOM. Nevýhodou je, že pokaždé musí být celý binární soubor zakódován pomocí base-64, což zdvojnásobuje cílovou velikost zprávy. Díky tomu, že SOAP je tvořen pomocí XML, tak při velké velikosti zprávy dochází k velkým požadavkům na prostředky cílového systému. (12)

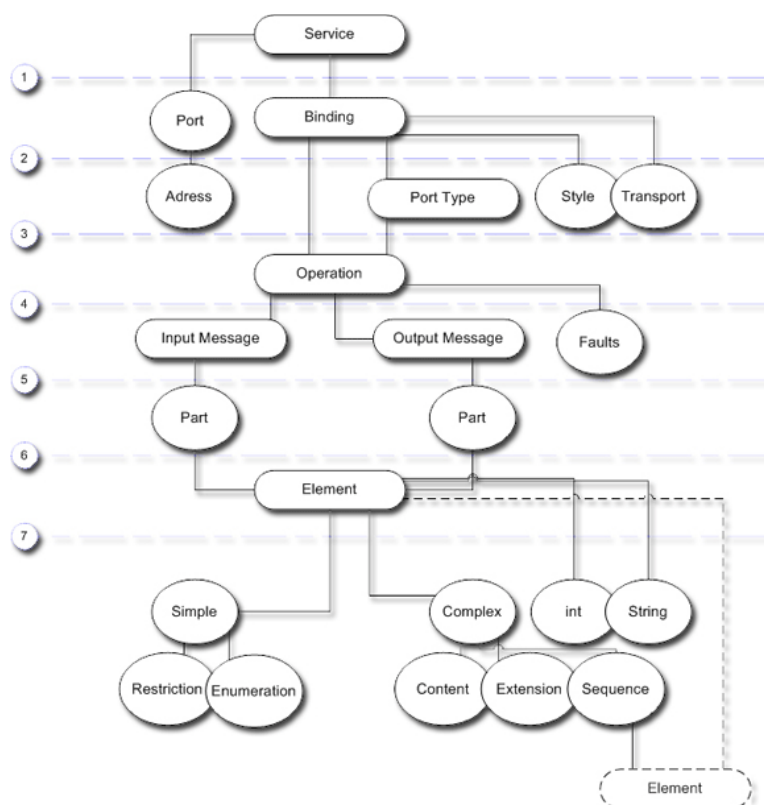
#### 4.2.1.3 Popis webové služby WSDL

Další součástí webových služeb je popisný jazyk WSDL<sup>27</sup>, který je ve formě XML dokumentu, jenž popisuje webovou službu. Jde o definici WS, která nabízí model popisující webové služby, respektive popisuje nabízené funkce a způsob jak se na ně dotázat. WSDL definuje služby jako kolekci síťových koncových bodů nebo portů. Abstraktní definice portů a zpráv je oddělená od konkrétního použití, díky čemuž umožňuje znovupoužití těchto definic. Port je definován s asociací na síťovou adresu s opakovaným použitím a zároveň s kolekcí portů definovaných jako služba. Zpráva jako taková je ve WSDL abstraktním popisem informací, která jsou vyměňována, a porty jsou abstraktní kolekcí podporovaných operací. Konkrétní protokol a specifikace určitého datového formátu pro konkrétní typ portu pak slouží opakovanému použití, kde operace a zprávy jsou navázány na konkrétní síťový protokol a formát zprávy. Tímto způsobem popisuje WSDL určité veřejné rozhraní pro samotnou webovou službu. Obsahem popisu jsou informace týkající se:

- Operací – tj. poskytovaných metod (funkcí) dané WS
- Definice struktury podporovaných zpráv (vč. datových typů)
- Definice koncových bodů – tj. způsobů volání WS
- Definice transportního protokolu + dostupných operací
- Protokol a adresa (<http://www.ws.org/search>)

---

<sup>27</sup> WSDL – Web Services Description Language



**Obr. 4-9: struktura WSDL**

WSDL jako takové není povinnou součástí definice webové služby, ale velmi usnadňuje její samotnou implementaci. V případě použití programovacích jazyků JAVA a .NET je možné definované WSDL pomocí programovacích nástrojů zpracovat a tyto už rovnou vytvoří samotné třídy a metody webové služby. Na programátorovi pak už zůstává samotné napojení na aplikaci, které webovou službu využívá. Toto platí hlavně (a v tuto chvíli možná pouze) pro verzi WSDL 1.1, která ovšem nikdy nebyla standardizována W3C, standardem se stala verze 2.0 (původně 1.2), která se od původní ovšem zdatně liší. (13)

#### **4.2.1.4 Rozšíření WS: specifikace WS-\***

Specifikace WS-\* je soubor rozšíření pro potřeby webových služeb. Samotný standard WS byl postupně rozšiřován o další specifikace označované pojmem WS-\*. Tyto standardy zejména doplňují WS o možnosti zabezpečení, autentizace, autorizace, routování, doručování a řady dalších rozšíření. Tato rozšíření WS-\* specifikace se mohou navzájem překrývat, lze skládat od jednoduchých služeb až po komplexní řešení integrace.

Některé WS-\* specifikace:

- WS-Security - příložení podpisů, šifrování (SAML, kerberos, x.509)
- WS-Policy - požadavky na zprávu (zabezpečení, tokeny, časová platnost, jiné)
- WS-Federation - distribuovaná autentizace (poskytování identit, SSO)

- WS-Addressing - směrování (routing), korelace (duplex), asynchronní zpracování (source/reply/fault endpointy)
- WS-ReliableMessaging - transakce, QoS
- A další

Specifikace WS-\* jsou opravdovým komplexním nástrojem, kombinací jednotlivých specifikací, tvůrce dosáhne velice komplexního systému zpracování a vyřizování zpráv. Ovšem jejich síla je zároveň i velkou slabinou, tyto specifikace mají velmi rozsáhlou a složitou dokumentaci a vytvoření komplexního portfolia služeb a jejich vazeb mezi sebou stojí velké množství času, jak na nastudování těchto specifikací, tak na samotný vývoj (ne nadarmo byla některými vývojáři překřtěna na WS-HELL). Na druhou stranu lze pomocí nich vytvořit komplexní B2B služby, které jsou jinak doménou korporátních společností (jako je HP, ORACLE a další), kdy tato řešení jsou taktéž složitá na implementaci a dokumentaci (ORACLE B2B pořadnou dokumentaci ani nemá, pokud nepočítáme blog vývojářů), jsou velmi drahá a jedná se o vlastní řešení uznávající vlastní standardy nebo standardy uzavřené (AS2).

#### 4.2.2 Zpracování zpráv pomocí WS

V této části se chci věnovat popisu způsobu, respektive formy komunikace mezi dvěma subjekty za využití webových služeb. Tyto formy jsou veskrze dvě, synchronní zpracování a asynchronní zpracování. Webové služby jsou v základním principu synchronní, tedy na vznesený požadavek (request) webová služba okamžitě poskytne odpověď (response). Ovšem i v rámci webových služeb lze realizovat asynchronní komunikaci a to hlavně z důvodů, kdy cílový systém na kterém je vystavená webová služba, není z pohledu zpracování dotazu systém poslední, ale zpracovávanou zprávu posílá dalším systémům, které obhospodařují vyřízení některých částí. Jedná se v abstraktním pohledu o určitý proces. Tyto asynchronní mechanismy byli do WS začleněny později, jde hlavně o specifikaci WS-Addressing a zde definované duplexní způsoby komunikace.

Při pohledu na synchronní zpracování lze říci, že je jednodušší na implementaci, ale náročnější na prostředky. Oproti tomu asynchronní zpracování je pravým opakem, jeho velkou výhodou je možnost distribuovat zátěž na straně poskytovatele (tj. serveru) i konzumenta (tj. klienta). Implementace je nicméně výrazně složitější (nutno implementovat korelační mechanismy).

Formu komunikace pomocí webových služeb lze modelovat v základu podle následujících definic:

1. Synchronní zpracování (request-reply):

A<->B (pouze B vystavuje WS)

2. Aktivní asynchronní zpracování (duplex):

A->B, B->A (obě strany vystavují WS, řešeno pomocí WS-Addressing za využití source/reply/fault endpointů)

### 3. „Pasivní“ asynchronní zpracování:

A<->B1 (požadavek), A<->B2 (odpověď), lze implementovat jako dvě synchronní volání z A na dvě služby B tj. B1 a B2.

Způsob 1 je řešení klasických webových služeb, kdy na jedné straně (A) je pouze volání webové služby (B), která okamžitě (synchronně) poskytne odpověď na požadavek, tím komunikace končí.

Způsob 2 je za využití rozšíření specifikace WS-Addressing (může být v kombinaci s dalšími WS-\*), kdy v samotné zprávě požadavku je definován zdrojový koncový bod (endpoint), koncový bod kam má být směřována odpověď a chybový koncový bod, pro případ nekorektního zpracování služby. Každá ze zpráv má uvedeno ID pro identifikaci zprávy a odpovědní a chybová zpráva má uvedeno dále ID původní zprávy s požadavkem, které slouží k párování se zprávou odpovědní. Toto řešení znamená, že obě komunikující strany musí mít vystaveny webové služby. Komunikace probíhá na principu kdy jedna strana (A) vznesle požadavek určený WS další straně (B), tato služba požadavek přijme a komunikace je ukončena. Po úspěšném zpracování volá dotazovaná strana (B) webovou službu strany čekající na odpověď (A) na koncový bod, který byl uvedený jako adresa určená pro přijetí odpovědi. V odpovědní zprávě je uvedeno ID zprávy a současně korelační ID původní zprávy. Poskytne odpověď a komunikace je ukončena. Toto řešení je elegantní, ale požaduje, aby obě komunikující strany měli vystavené webové služby se specifikací WS-Addressing, což v případě kdy konzumentem je několik desítek či stovek klientských systémů někdy není dost dobře možné, protože každý z těchto systémů by musel implementovat cele složitě řešení.

Způsob 3 je jakýmsi hybridem, který klade celou váhu komunikace na systém poskytovatele. Jedná se o způsob, kdy za využití dvou synchronních služeb modelujeme asynchronní zpracování. Komunikace probíhá následovně, dotazující strana (A) volá WS poskytovatele (B1), ten požadavek přijme a odpoví, součástí odpovědi je ID (např. token), komunikace je ukončena. Na straně poskytovatele dochází ke zpracování. Následně dotazující se strana (A) zavolá další WS poskytovatele (B2), předá ID, které obdržela a získá odpověď, ta může být ve formě odpovědi na původní požadavek, ale také může obsahovat informaci, že požadavek nebyl doteď zpracován. Toto řešení umožňuje na straně dotazujícího relativně snadnou implementaci, ovšem za cenu, že se musí třeba i několikrát dotazovat WS (B2), zda byl její požadavek zpracován. Na straně poskytovatele to nicméně znamená o dost složitější implementaci, jelikož břímě komunikace je přeneseno na jeho stranu. V řadě případů jde ovšem o jediný způsob jak komunikovat asynchronně s širší skupinou klientů, kteří nejsou schopni implementovat duplexní webové služby.

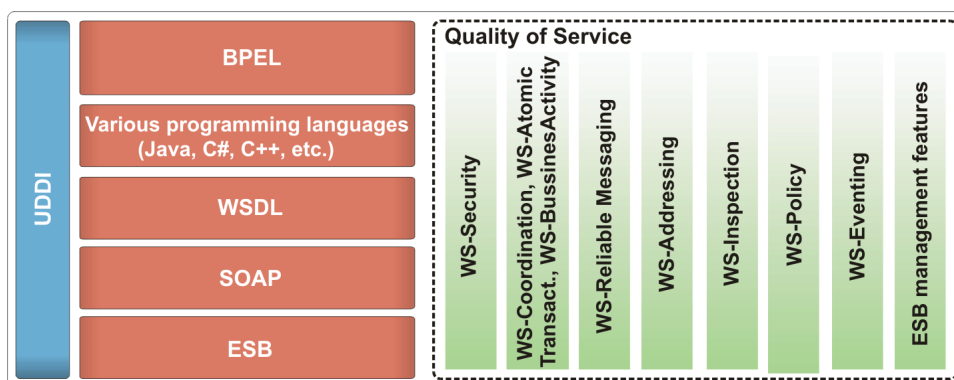
Asynchronní zpracování je v současnosti zejména doménou messagingových technologií a B2B protokolů (JMS, AQ, MSMQ, AS2 ...), které ovšem pro

použití skupinou většího množství malých systémů nejsou řešením, slouží pro propojení velkých informačních systémů. (16) (17)

### 4.2.3 Využití WS v SOA architektuře

SOA<sup>28</sup> tj. servisně orientovaná architektura je metoda vývoje aplikací založená na principu využívání atomických služeb. Tyto služby jsou poskytovány systémy a jako takové vykonávají určitou pevně definovanou činnost. V práci se o nich zmiňují z důvodů, že využití SOA jako návrhu architektury umožňuje komplexní modelování služeb a jejich využívání. Pro účely EZD je nesmírně důležité, aby tento systém byl navržen jako co nejvíce znovupoužitelný a vzniklé zásahy probíhali na úrovni služby, které se změna dotýká a nezasahovali do systému jako celku.

Jednotlivé služby poskytované v rámci SOA architektury jsou bezstavové, deterministické, volně propojené. Skládáním definovaných služeb je realizován určitý proces, který můžeme přirovnat k počítačovému programu. Tento proces se nazývá business proces.



Obr. 4-10: struktura SOA architektury

V rámci architektury SOA jsou často používány webové služby, nejedná se ale o jediný způsob volání služeb daného systému (alternativy - JMS, AQ, HTTP, FTP,...). V rámci SOA mohou být používány systémy využívající různé technologie a formy komunikace (WIN/Unix, XML/CSV, AS2/EDI, ebXML/HL7 ...). Jelikož ne každý systém umožňuje vystavení webových služeb, je vhodné použít sjednocující prvek – ESB (Enterprise Service Bus). ESB slouží k propojení různých systémů do společné komunikační sběrnice, lze ho přirovnat k jakékoliv jiné sběrnici, tak jak ji známe z HW architektury. V rámci ESB se definují jednotlivé pravidla pro každou službu a celek se potom chová jako nástroj, kde každá služba má definované posloupnosti zpracování, volání další služeb. V rámci ESB lze vystavit WS, poskytující službu vybraného systému, který je do ESB připojen. Díky tomu webové služby, přestávají být chápány jako

<sup>28</sup> SOA – Service Oriented Architecture

jednorázové procesy a stávají se univerzálním prostředkem volání aktivit mezi jednotlivými systémy při vytváření SOA aplikací.

Způsoby práce s webovými službami v rámci architektury SOA jsou dva, metodě práce s nimi se říká kompozice webových služeb v SOA.

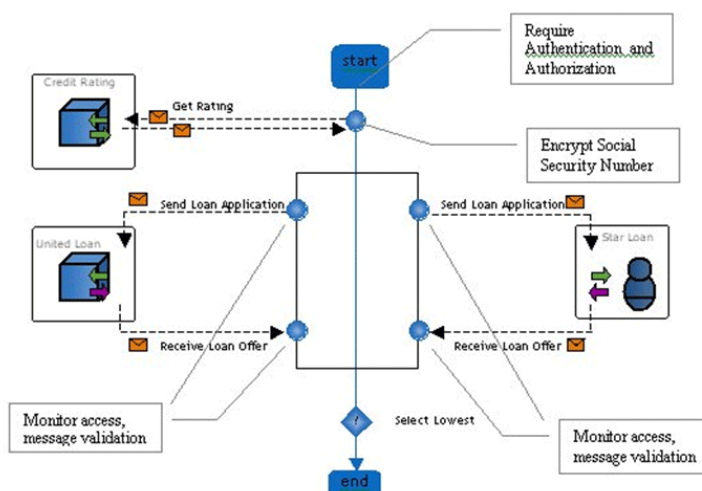
Kompozice webových služeb v SOA formou:

- WS orchestrace
- WS choreografie

Tyto názvy nemají nic společného s terminologií informačního inženýrství, jsou zavádějící, nicméně se ujal a jsou používány.

WS orchestrace – WS-BPEL, vychází z principu, kdy existuje jeden koordinátor (může být další WS) webových služeb, který zajišťuje komunikaci jak s vnějšími tak vnitřními službami a zároveň se i stará o řízení a návaznosti volání těchto služeb. V rámci orchestrace je jednoznačně řízeno volání jednotlivých služeb, výsledkem je tzv. business proces, tímto se zabývá tzv. Business Process Modeling (BPM).

BPEL <sup>29</sup>jako takový je XML jazyk, za vývojem tohoto standardu stojí (zastřešuje) organizace OASIS. Jeho hlavním předmětem činnosti je poskytování notace pro specifikaci podnikových procesů založených na WS. Již z této definice je zřejmé, že zaměřením pouze na WS je jazyk BPEL zaměřen pouze na část SOA. Pomocí BPEL je velmi dobře možné realizovat dlouhé asynchronní byznys transakce včetně kompenzačních. Jazyk zvládá řízení toku dat (proměnné, korelace, cykly, větvení, spojení) i ošetření chyb a výjimek. BPEL servery obecně jsou stavové a umožňují běh dlouhodobých transakcí. Standard funguje nad WS (WSDL).



Obr. 4-11: ukázka WS orchestrace procesu v BPEL

<sup>29</sup> BPEL - Business Process Execution Language

WS choreografie – WS-CDL (W3C), je postavená na principu spolupráce mezi WS. Jednotlivé WS jsou si navzájem rovny (žádný koordinátor), mohou se volat navzájem za situace, kdy každá zúčastněná služba přesně ví, kdy se má spustit a s kým má komunikovat, přičemž tato volání jsou stavová a mohou být dlouho trvající. Všechny služby účastníci se choreografie musí vědět o procesu, jeho operacích, zprávách a načasování výměn zpráv. (16) (17)



## 5 Analýza systému EZD

V této části své diplomové práce jsem se soustředil na mnou zvolený postup návrhu a vývoje systému EZD. Zvolené postupy jsem tvořil postupným rozkladem shora – dolů, kdy se nejdříve soustředím na vytvoření základní modelové situace a popsání, respektive znázornění jednotlivých operací probíhajících v systému. Dále se věnuji vytvoření a popsání návrhu databázového modelu, který zobrazuje navrhované rozdělení entit a jejich vazby mezi nimi.

### 5.1 Use Case digramy systému EZD

Pro vytvoření modelu chování systému jsem použil diagramy Use Case z modelovacího jazyka UML<sup>30</sup> 2.0. Tento diagram ukazuje funkce poskytované systémem, z pohledu použití uživateli/subjekty s ním operujícími (systém je taky uživatel), jejich cíle jsou zastoupeny případy použití a jakékoliv závislosti mezi těmito případy užití. Jedná se o relativně jednoduchý nástroj, ovšem s velkou výpovědní hodnotou. Jako takový je velmi vhodný k popisu elementárních situací a operací v rámci systému, kdy jasně a přehledně ukazuje zvolený postup, lze jej jednoduše upravovat a zároveň je čitelný a přehledný i pro člověka bez technických znalostí. Veškeré modely jsem vytvořil pomocí nástroje Enterprise Architect verze 7.

Dále v textu se soustředím pouze na některé modelové situace, jako jsou:

- Přístup klientů a zdravotnických pracovníků přes webové rozhraní
- Správa EZD klientem
- Správa EZD zdravotním pracovníkem
- Přístup zdravotnických pracovníků a zařízení pomocí B2C brány
- Zápis a nahlížení nad lékařskými zprávami

Jedná se pouze o základní soubor modelových situací v rozsahu EZD je třeba popsat všechny situace, které v systému mohou nastat, detailní analytická dokumentace za pomocí Use Case je uvedena v příloze této diplomové práce.

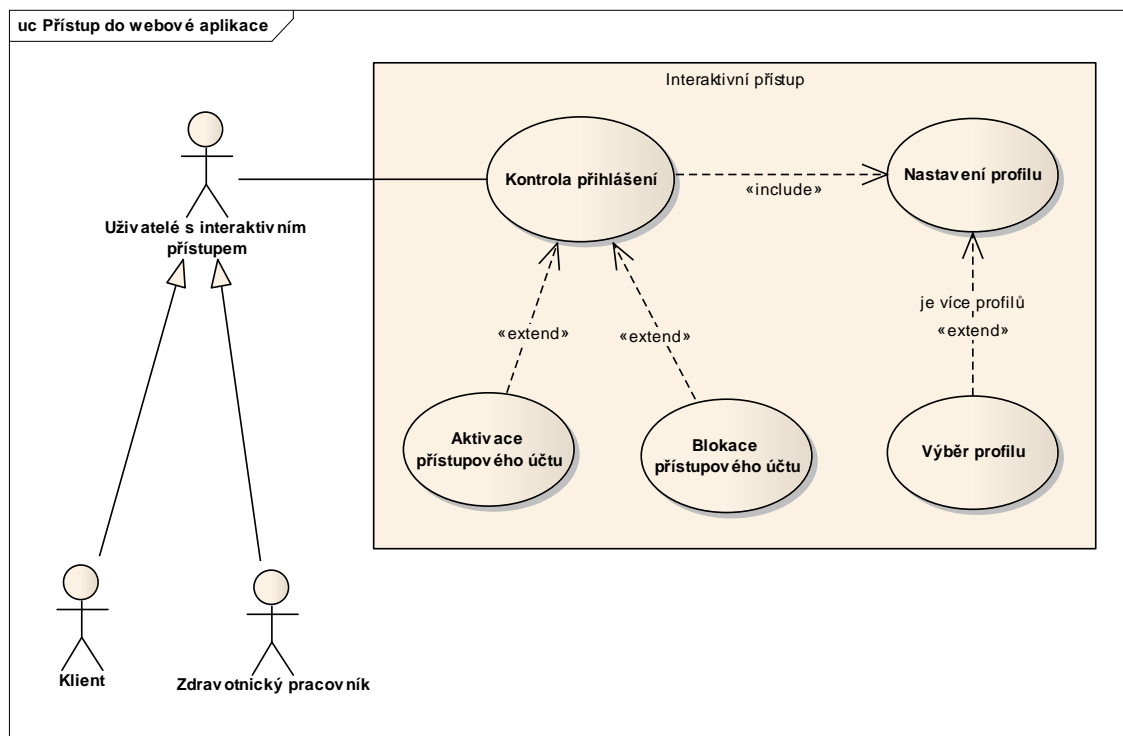
#### 5.1.1 Přihlášení klienta, zdravotnického pracovníka

Tento diagram zobrazuje scénář použití přihlášení a autorizace do aplikace EZD. Autorizace uživatelů v rámci aplikace EZD probíhá v následujících krocích:

- Autorizace uživatele na základě informací ze systému
- Aktivace přístupového účtu pomocí PINu/certifikátu, pokud je neplatný tak blokáce při určitém počtu neúspěšných přístupů
- Nastavení profilu uživatele (případně volba profilu)

---

<sup>30</sup> UML - Unified Modeling Language

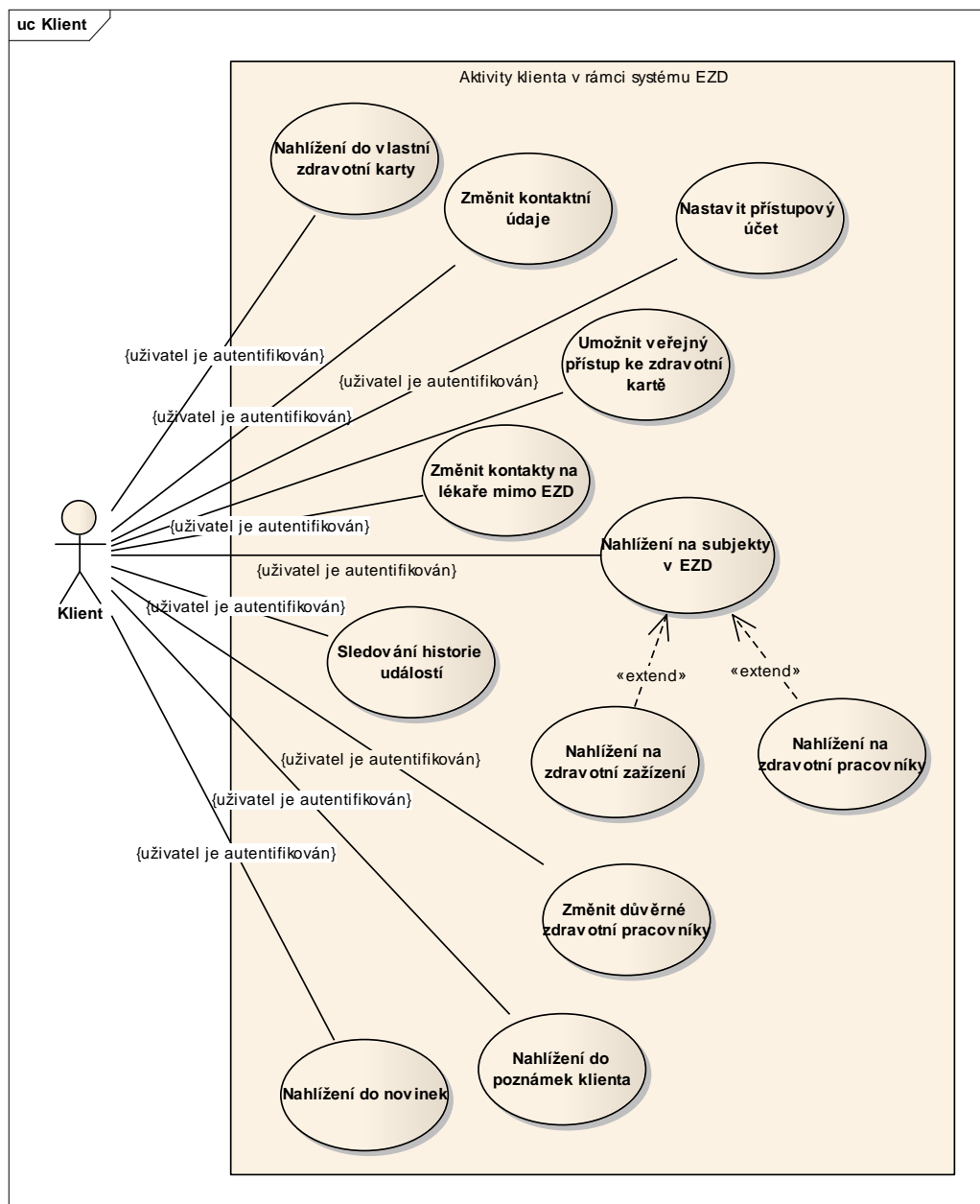


Obr. 5-1: přístup klienta do webového rozhraní EZD

### 5.1.2 Správa EZD klientem

Zde je zobrazen scénář, či spíše abstrakt práce uživatele ve webovém rozhraní se systémem EZD. Popisuje jednotlivé možnosti práce v systému jako takovém v pohledu přístupu do částí systému určených klientu, správu kontaktních údajů, profilu, historii, prohlížení zdravotní karty a vlastního nastavení práv přístupu do zdravotní karty klienta zdravotním pracovníkům. Ve všech případech se přístup do EZD řídí procesem přihlášení a klient pro práci s EZD musí být autentizován.

Nastavení práv na přístup do zdravotní karty klienta pro zdravotní pracovníky je možný několika způsoby. Jedním je možnost vystavení souhlasu s veřejným přístupem, kdy klient souhlasí s prohlášením, že všichni jeho ošetřující zdravotní pracovníci mohou nahlížet na jeho zdravotní data. Další možností je vytvoření množiny zdravotních pracovníků, označovanou v systému jako „důvěrní lékaři“, kdy těmto zdravotním pracovníkům je umožněno nahlížet do zdravotních záznamů klienta. Praktický lékař se stává automaticky důvěrným. Případně může klient zamezit přístup všem zdravotním pracovníkům a poskytovat souhlas pouze v jím zvolených případech, přes rozhraní zdravotního pracovníka. Ovšem každý zdravotní pracovník má právo zápisu zdravotních informací do EZD klienta.

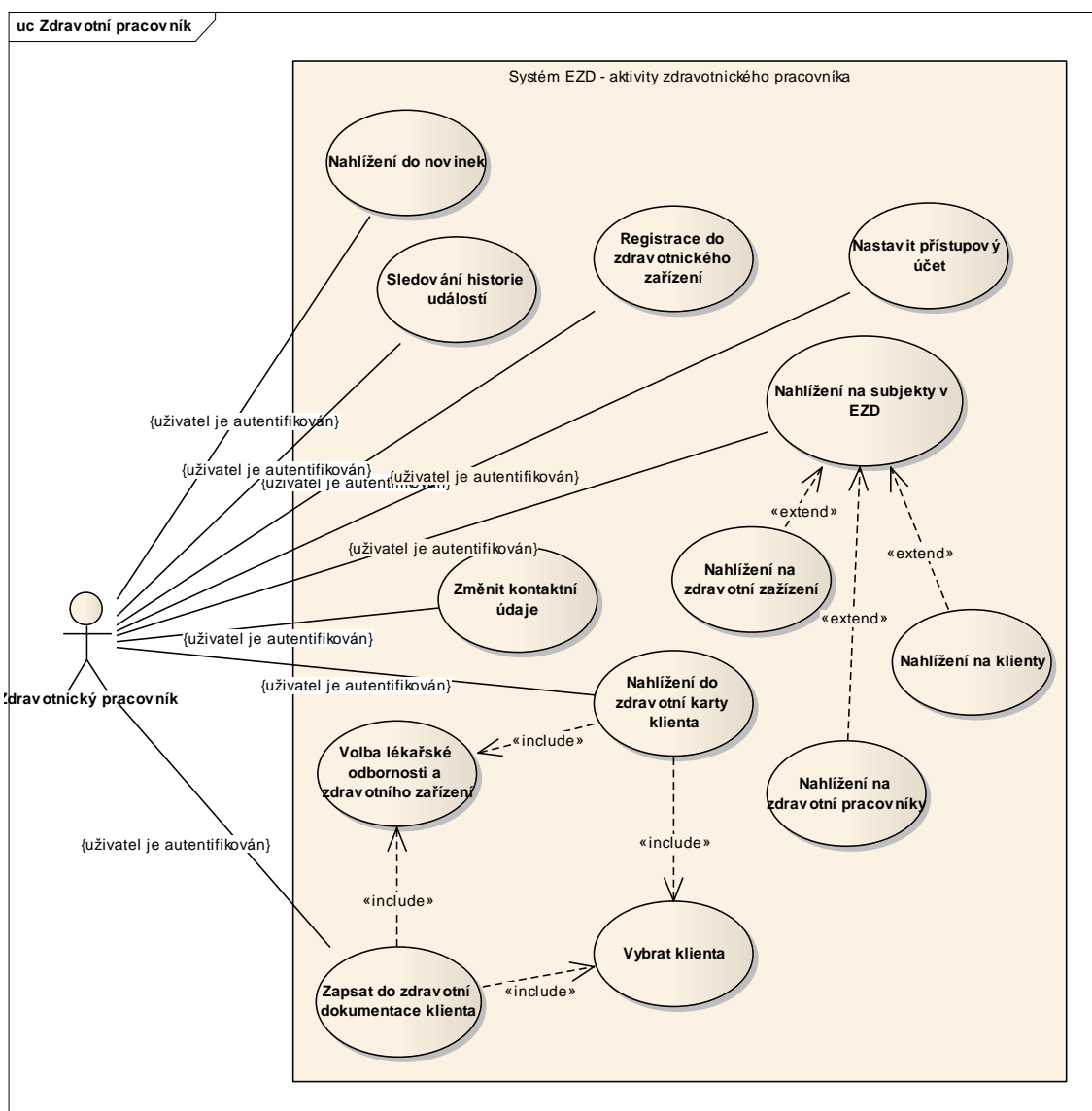


Obr. 5-2: aktivity klienta v systému EZD

### 5.1.3 Správa EZD zdravotním pracovníkem

Scénář zde popsaný zobrazuje aktivity zdravotního pracovníka v systému EZD. Popisuje podobné činnosti jako scénář Obr. 5-2: aktivity klienta v systému EZD ovšem z pohledu systému se jedná a rozdílný aplikační model. Většina modulů je unikátních pro zdravotního pracovníka a zobrazuje informace směřované primárně pro účely zdravotnických pracovníků. Taktéž umožňuje správu kontaktních údajů, profilu, nahlížení do historie. Navíc zde je správa zdravotní karty klienta. Stejně jako v případě klienta, je zdravotnický pracovník nucen se autentizovat pro práci se systémem EZD.

Práce se zdravotní kartou klienta probíhá následujícím způsobem, zdravotní pracovník si zvolí klienta, se kterým chce pracovat (vyhledá podle čísla pojištěnce, zvolí v seznamu) a zobrazí se mu zdravotní karta klienta, podle nastavení práv klientem. Může tedy zapsat záznam, případně i nahlížet do zdravotních informací. Pokud klient má zakázáno nahlížení zdravotních pracovníků do své zdravotní karty, je možnost v této části autorizovat zdravotního pracovníka k nahlížení do zdravotní karty, zadáním svých autentizačních údajů.



Obr. 5-3: aktivity zdravotního pracovníky v systému EZD

#### 5.1.4 Vstup zdravot. zařízení/pracovníka pomocí B2C brány

Tato brána bude sloužit k automatické komunikaci z informačních systémů lékařů (AIS), nemocnic (NIS), laboratorních zařízení (LIS) a lékáren. Je koncipována jako soubor webových služeb.

Do WS brány (B2C) mají přístup pouze uživatelé splňující následující kritéria:

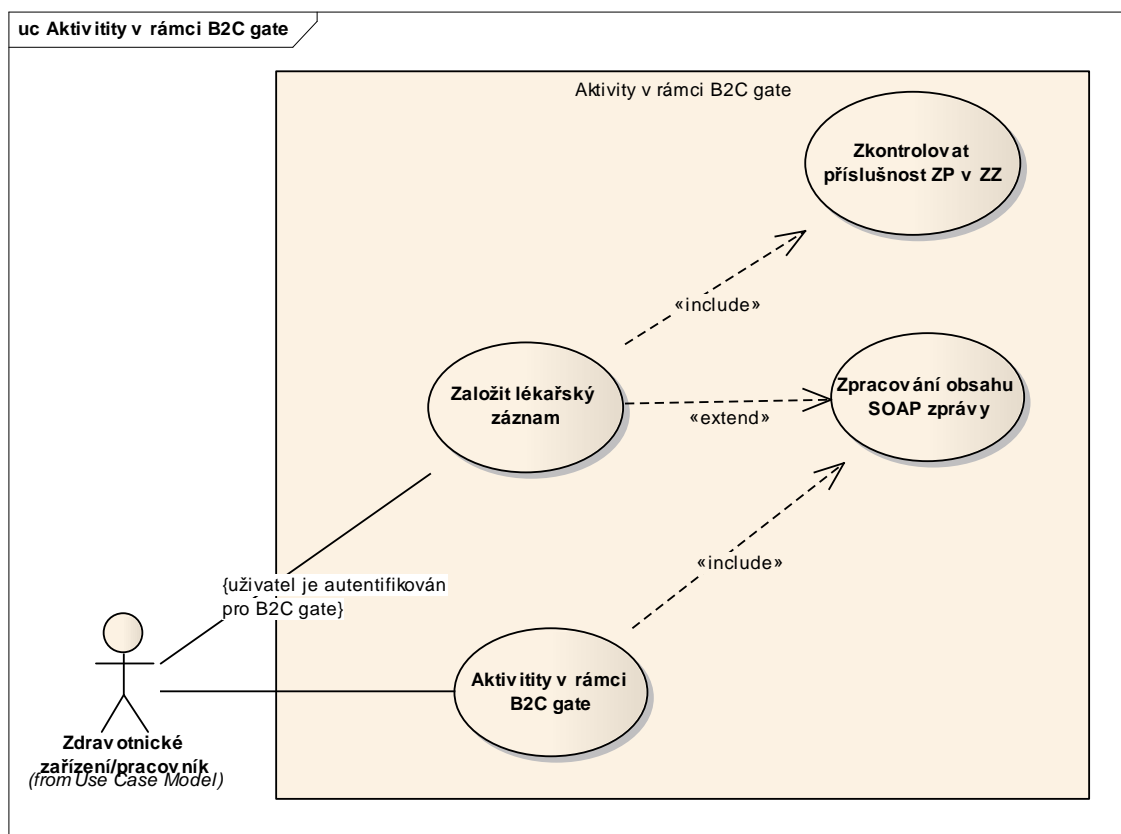
- Aktivní účet EZD
- Aktivní profil EZD
- Profil EZD musí podporovat přihlášení do WS brány aplikace EZD

Přihlašovat se mohou pouze uživatelé s některým z následujících aktivních profilů:

- Zdravotnické zařízení,
- Zdravotnický pracovník v existujícím zdravotnickém zařízení

Zpracování samotných lékařských zpráv, které jsou obsaženy v dané SOAP zprávě musí splňovat požadavky na datový formát:

- DASTA 3.xx.xx
- DASTA 4.xx.xx



Obr. 5-4: přístup zdravot. zařízení/pracovníka pomocí B2C brány

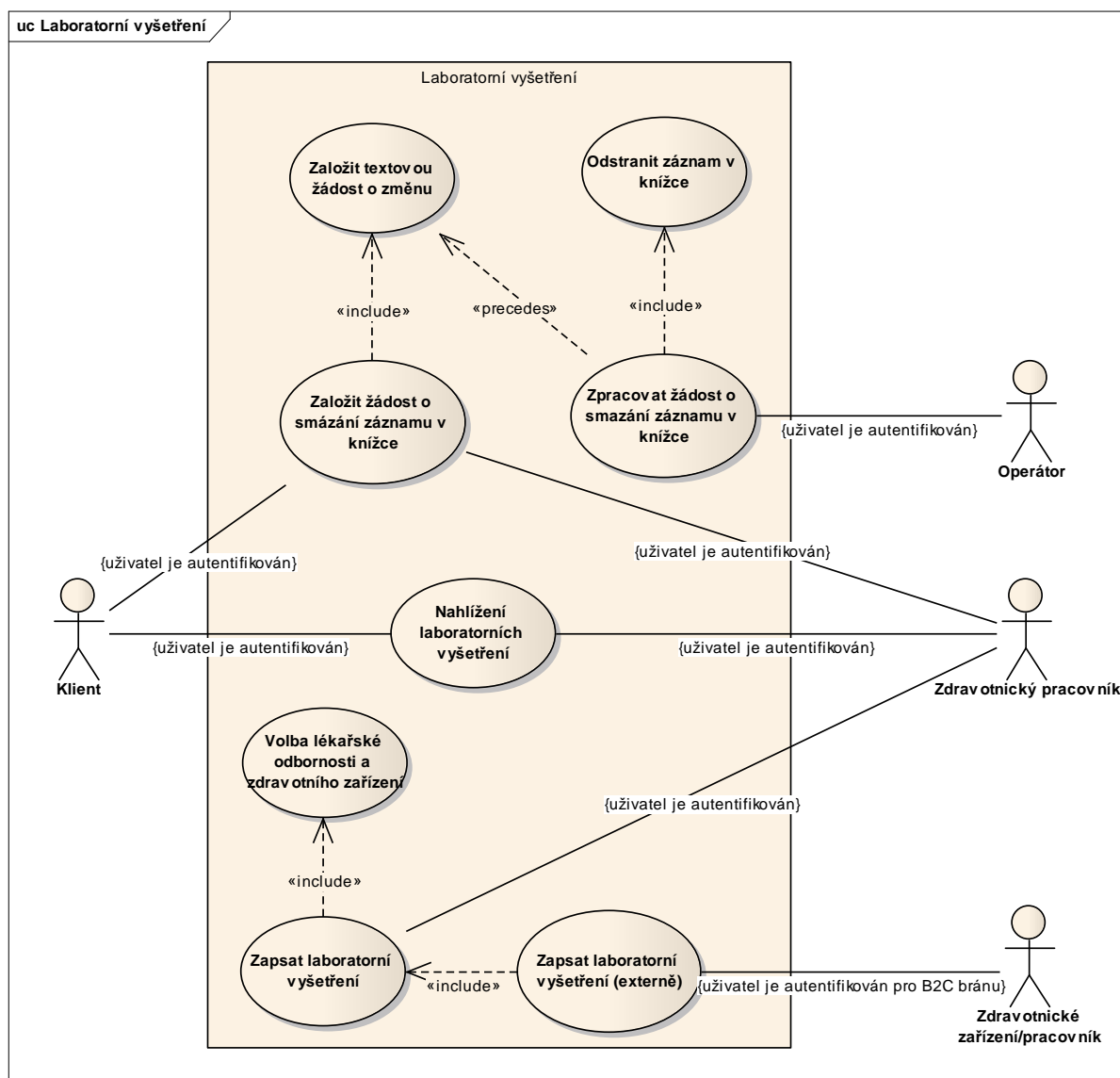
### 5.1.5 Zápis a nahlížení nad lékařskými zprávami

Popsaný scénář ukazuje způsob práce se zdravotním záznamem, jako ukázka bylo zvoleno laboratorní vyšetření. Zobrazení je z pohledu zdravotní zprávy, kdy

jsou zobrazeny jednotlivé akce a operace se zdravotním záznamem, v tomto případě s laboratorním vyšetřením.

Uživatelské role pro specifické činnosti se zdravotním záznamem jsou následující:

- Klient – nahlížení a žádost a smazání zdravotního záznamu
- Operátor – vyřizuje žádost klienta/zdravotního pracovníka o smazání zdravotního záznamu
- Zdravotnický pracovník – nahlížení a zápis, smazání (pouze záznamu kde je autorem) zdravotního záznamu
- Zdravotnický pracovník (v zařízení) – zápis zdravotního záznamu



Obr. 5-5: zápis a nahlížení nad zdravotním záznamem

## 5.2 Databázový model aplikace – zdravotní zprávy

Při přípravě tvorby datového modelu, jsem se rozhodl pro vytvoření části zabývající se způsobem ukládání a práce se zdravotními záznamy. Jsou zde naznačeny vazby na další entity, jako je klient, zdravotní pracovník a podobně, ale detailně zpracované jsou pouze entity týkající se zdravotní zprávy, vazby mezi zdravotnickým pracovníkem a zdravotnickým zařízením.

Databázový model, respektive její část (správa zdravotních zpráv) používá jednotných a pevně definovaných názvových konvencí. Při vytváření konvencí pojmenování objektů v datovém modelu jsem kladl důraz na maximální zjednodušení a současně na vysokou čitelnost názvů objektů v nich použitých.

### 5.2.1 Použité kódové konvence

#### 5.2.1.1 Tabulky

Název tabulek se skládá z 3písmenného prefixu psaného velkými písmeny, následovaného podtržítkem a dále vlastním názvem tabulky (např. MRE\_Record).

Pravidla pro prefix:

- 3 znaky
- Všechna písmena prefixu jsou velká
- Významově určuje oblast problému, které tabulka náleží (více tabulek může sdílet stejný prefix)

Pravidla pro název tabulky:

- Pojmenování pomocí PascalCase
- Pojmenování pomocí podstatného jména v jednotném čísle (tedy správně je MRE\_Record a ne MRE\_Records)

#### 5.2.1.2 Sloupce tabulky obecně

Podobně jako tabulka i sloupce tabulky se skládají z 3písmenného prefixu následovaného podtržítkem a dále názvem sloupce (např. REC\_Content, RFL\_Data).

Pravidla pro prefix

- 3 znaky
- Všechna písmena prefixu jsou velká
- Významově vychází z názvu tabulky pokud možno maximálně jej asociuje/připomíná.

Pravidla pro pojmenování sloupce tabulky

- Pojmenován pomocí PascalCase

- V názvu sloupce se neopakuje název tabulky (kromě odůvodněných případů)

### 5.2.1.3 Primární klíče

Primární klíč tabulky, který je tvořen automatickou identitou nebo číselnou hodnotou obecně (int, bigint) je pojmenován se suffixem \_ID (identifier). Naproti tomu primární klíče nad textovými sloupci se označují jako tzv. „kódy“ a pro jejich pojmenování je zvolen suffix \_CID (code identifier). Datovým typem kódů je vždy varchar(3).

### 5.2.1.4 Cizí klíče (referenční integrita)

Sloupce reprezentující odkazy na primární klíč jiné tabulky jsou pojmenovány suffixem \_FK (foreign key).

V případě, že je sloupec s cizím klíčem zároveň primárním klíčem tabulky (časté např. u ISA<sup>31</sup> vztahů, odvozeno z anglického "is a" např. "Lukas is a human".) je pojmenován podle pravidla pro primární klíč (suffix \_ID na konci). Toto pravidlo však neplatí, pokud je cizího klíče použito v tzv. složeném primárním klíči. Pojmenování se pak řídí dle obecného pravidla pro pojmenování cizích klíčů.

### 5.2.1.5 Indexy

Pojmenování indexů nad sloupci v tabulce se liší dle toho, zda vytváří primární klíč, unikátní index (alternativní klíč – viz samotné konvence) nebo „klasický“ index.

V případě klasického indexu je název složen z 2písmenného prefixu IX (index), podtržítkem a prefixem sloupců tabulky a dále názvy jednotlivých sloupců indexu včetně suffixů (pokud jsou suffixy sloupců v indexu navzájem shodné lze jednotlivé suffixy vynechat a spojit je na konci názvu do jednoho).

Pravidla pro prefix:

- prefix IX\_
- 3 znaky z prefixu sloupců tabulky

Pravidla pro název indexu:

- Pojmenován pomocí PascalCase
- Obsahuje vždy názvy všech sloupců obsažených v indexu dle jejich pořadí
- Názvy sloupců v indexu neobsahují prefixy

---

<sup>31</sup> ISA vztah je vztah vypovídající o hierarchickém uspořádání světa



#### 5.2.1.6 Alternativní klíče

Název alternativního klíče tabulky se řídí stejnými pravidly jako indexy s tím rozdílem, že alternativní klíč používá prefix AK (alternate key).

Pravidla pro prefix:

- Prefix AK\_
- Dále viz pojmenování indexů

#### 5.2.1.7 Primární klíče tabulek (indexy)

Primární klíče tvořené číslem (identitou nebo ručně) jsou definovány jako 64bitové integery (bigint). Pojmenování indexů primárních klíčů je shodné s pojmenováním alternativních klíčů s tím rozdílem, že pro primární klíče je používá prefix PK\_ (primary key).

Pravidla pro prefix:

- Prefix PK\_
- Dále viz pojmenování alternativních klíčů

### 5.2.2 Lékařské záznamy a zdravotní dokumentace

Vzhledem k celkovému rozsahu datového modelu pro EZD, jsem se rozhodl popsat návrh pouze pro oblast týkající se samotného uchovávání zdravotních zpráv. Tento navrhovaný model je určitou abstrakcí, kterou lze fyzicky realizovat, ale jako takový není optimalizovaný pro praktické použití. Je navržen se splněním metodiky 3NF<sup>32</sup>.

Oblast datového modelu aplikace EZD, která pokrývá evidenci lékařských záznamů, je jádrem vlastní elektronické zdravotní dokumentace (EZD).

Manipulace se zdravotními záznamy (ZD) lze v rámci aplikace EZD shrnout zhruba následovně:

- Zpracování ZD z externího datového zdroje (zdrojem je buď ZZ či ZP)
- Vkládání a úprava ZD v rámci aplikace EZD (uživatelské rozhraní)
- Přístup k informacím ZD z hlediska aktérů EZD (pacient, ZP, ZZ)

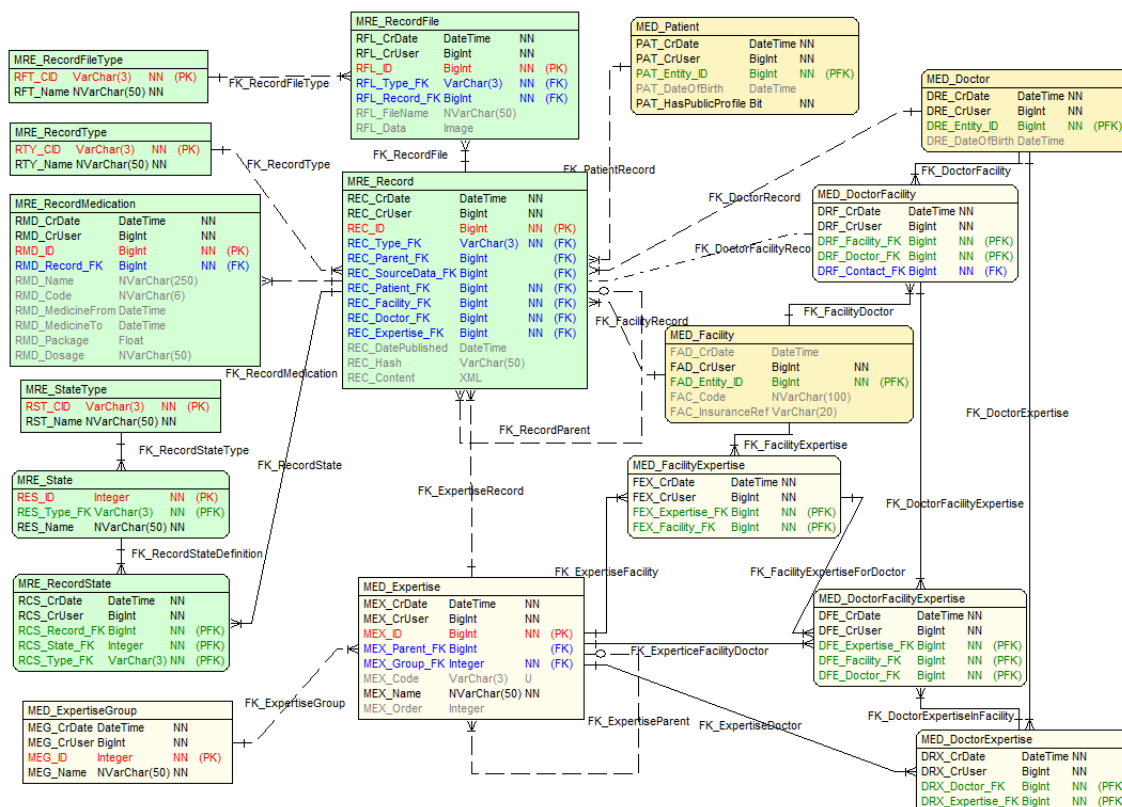
Výše uvedené způsoby přístupů k EZD samozřejmě nejsou konečné. Jejich vlastní specifikace a principy jsou však hlavně záležitostí aplikační vrstvy a proto se jejich problematice nebudou věnovat.

---

<sup>32</sup> 3NF – třetí normální forma

V následujících kapitolách je podrobněji přiblížen způsob zpracování lékařských odborností, jejich rozdělení do skupin, vlastní lékařská zpráva a její stavy a dále přiložené informace lékařských zpráv.

Datový model evidence lékařských zpráv včetně všech závislých objektů je uveden v přiloženém obrázku.



Obr. 5-6: schéma lékařských zpráv a závislých záznamů (ROE)

### 5.2.2.1 Lékařské odbornosti

Lékařské odbornosti zastávají v rámci aplikace EZD princip množin oprávnění daného ZP resp. ZZ na část EZD. Odbornost určuje, jaké druhy zdravotních zpráv a akce nad nimi jsou dostupné ZP a ZZ. Bližší specifikace konkrétních oprávnění plynoucích z vybraných odborností nejsou v datovém modelu evidovány, jejich zpracování by se mělo plně odvíjet od zvolené implementace v aplikační vrstvě.

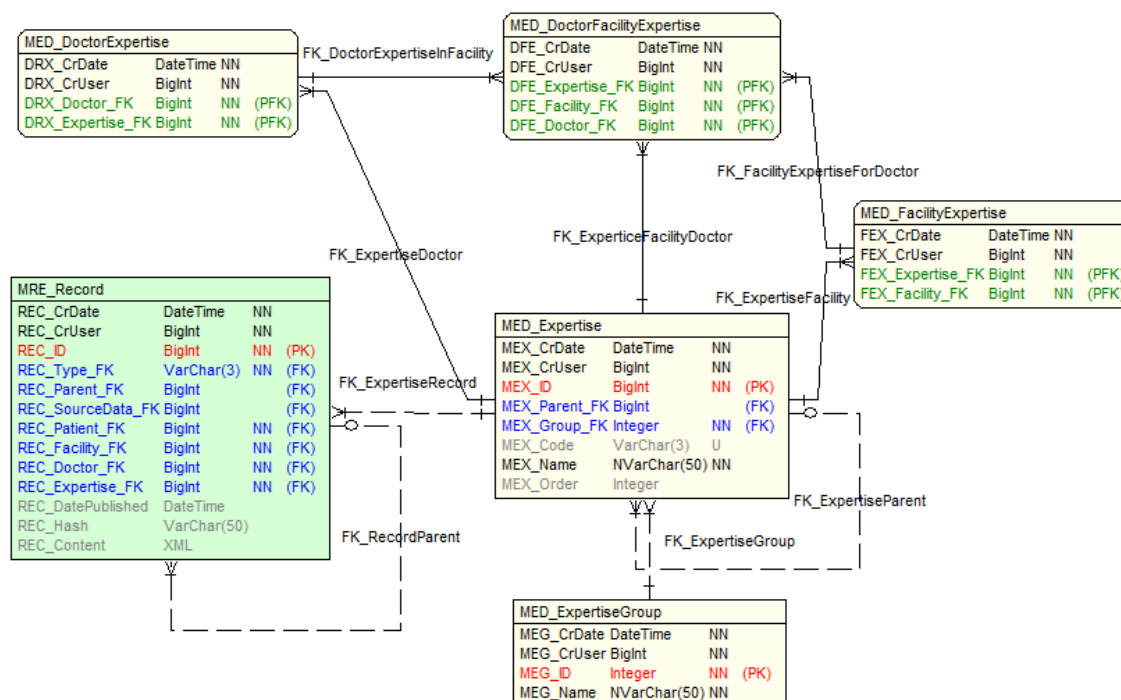
Datový model umožňuje evidenci lékařských odborností na úrovni:

- Zdravotního pracovníka (např. „odbornost lékaře“)
- Zdravotního zařízení

Zatímco v případě ZP vyjadřují odbornosti množinu „dovedností“, které ZP v rámci poskytování lékařské péče nabízí, v případě ZZ se jedná o druh lékařské péče, která je v daném zařízení dostupná. Příklady lékařské odbornosti jsou například:

- Praktický lékař - 001
- Otolaryngologie (ORL) - 701
- Toxikologická laboratoř - 814

Definice kódů a pojmenování lékařských odborností jde mimo rámec aplikace EZD. Zdrojem dat je v tomto směru odpovídající legislativní orgán dané země (například Ministerstvo zdravotnictví ČR).



Obr. 5-7: schéma lékařských odborností

#### 5.2.2.1.1 Skupiny odborností - [MED\_ExpertiseGroup]

V tabulce [MED\_ExpertiseGroup] jsou uloženy definice skupin lékařských odborností. Skupiny odborností slouží ke sdužování lékařských odborností do logických celků.

Na rozdíl od odborností, které jsou definovány na základě údajů legislativních orgánů dané země, jsou skupiny odborností „produktem“ aplikace IZK. S vlastní definicí odborností nemají skupiny odborností, kromě svého sdužujícího charakteru, žádnou souvislost.

Tabulka 5-1: struktura skupin zdrav. odborností [MED\_ExpertiseGroup]

Název sloupce	Typ	Pozn.	Popis
MEG_CrDate	datetime		Datum založení záznamu
MEG_CrUser	bigint		Autor záznamu (Id uživatele, není cizí klíč)
MEG_ID	bigint	PK	Id skupiny lékařské odbornosti
MEG_Name	nvarchar(50)		Název skupiny lékařské odbornosti

### 5.2.2.1.2 Odbornost - [MED\_Expertise]

Tabulka [MED\_Expertise] slouží pro ukládání definic lékařských odborností. Každá lékařská odbornost je povinně definována názvem a kódem. Oba tyto údaje jsou přiřazeny odpovídajícím legislativním orgánem. Kód lékařské odbornosti je univerzálním identifikátorem všech odborností. Tento údaj je platný vždy v rámci dané země.

**Tabulka 5-2: struktura odborností [MED\_Expertise]**

Název sloupce	Typ	Pozn.	Popis
MEX_CrDate	datetime		Datum založení záznamu
MEX_CrUser	bigint		Autor záznamu (Id uživatele, není cizí klíč)
MEX_ID	bigint	PK	Id lékařské odbornosti
MEX_Parent_FK	bigint	FK	Id nadřazeného záznamu v [MED_Expertise].[MEX_ID]
MEG_Group_FK	int	FK	Id skupiny odbornosti v [MED_ExpertiseGroup].[MEG_ID]
MEX_Code	varchar(3)		Kód odbornosti (přiřazen externě)
MEX_Name	nvarchar(50)		Název lékařské odbornosti
MEX_Order	int		Pořadí odbornosti v rámci nadřazeného záznamu

### 5.2.2.1.3 Odbornosti ZP a ZZ

Obecně jsou odbornosti ZZ uloženy v tabulce [MED\_FacilityExpertise], pro evidenci odborností ZP se používá tabulek [MED\_DoctorFacilityExpertise] a [MED\_DoctorExpertise].

### 5.2.2.2 Lékařské zprávy a přiložené informace

Lékařská zpráva (LZ) je základním stavebním kamenem EZD. V pojetí aplikace EZD se za lékařskou zprávu považujeme jakýkoliv výkon ZP, který je uložen do tabulky [MED\_Record] a závislých objektů, čímž se z ní stává zdravotní záznam (ZD).

Datový model aplikace EZD umožňuje evidenci různých druhů LZ. Obecně podporuje datový model následující objekty LZ a vztahy mezi nimi:

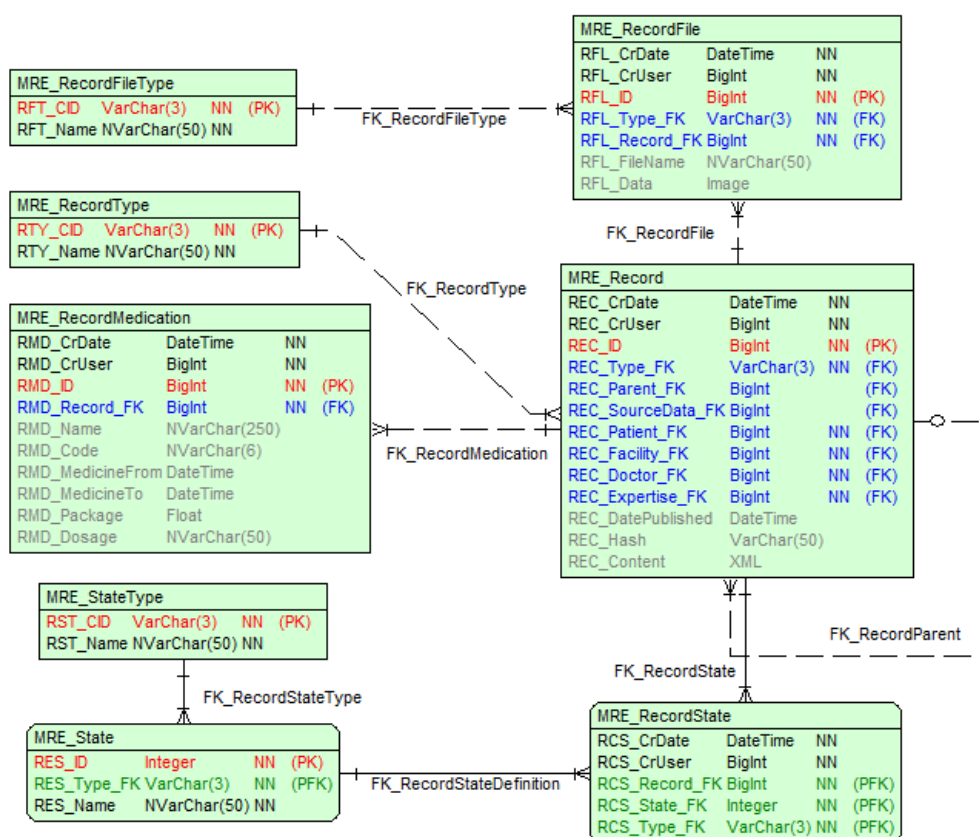
- Lékařská zpráva různých druhů
- Doplnující lékařská zpráva (rovněž také podřízená nebo související zpráva)
- Léky (předpis jen jinou formou lékařské zprávy)
- Přiložené soubory

Každou LZ obecně definují následující údaje:

- Datum vydání
- Typ zprávy (tj. zda se jedná o předpis, anamnézu, očkování atp.)
- Autor zprávy (tj. ZP, který zprávu vydal, zpravidla ošetřující lékař)
- Adresát zprávy (tj. pacient, kterého se zpráva týká)
- Odbornost, v rámci které byla zpráva vydána

- Vazba na nadřazenou zprávu (pokud lze aplikovat)

Vzhledem k citlivosti údajů prezentovaných v databázi EZD a dále oboru platnosti LZ, který je definován datem vydání zprávy, jsou v jednotlivých objektech LZ evidovány všechny historické záznamy. Důsledkem je, že neplatná (prošlá) data LZ se neodstraňují, či případně je nelze po jejich založení nijak upravovat. Každá operace upravující data se záznamem LZ vyvolá vždy založení nového záznamu a to bez ohledu na rozsah změn realizovaný v rámci uvedené operace. Již neplatné LZ lze od platných záznamů rozlišit pouze na základě data vydání a nastaveného příznaku se stavem zprávy.



Obr. 5-8: schéma lékařské zprávy

### 5.2.2.2.1 Typy lékařské zprávy - [MED\_RecordType]

V tabulce [MED\_RecordType] jsou uloženy druhy lékařských zpráv podporovaných aplikací EZD. Na úrovni datového modelu má volba typu lékařské zprávy vliv na obsah sloupce [MED\_Record].[REC\_Content]. V hodnotě tohoto sloupce je uložen obsah vlastní lékařské zprávy ve formě XML fragmentu. Struktura XML fragmentu uloženého v tomto sloupci se liší právě s ohledem na zvolený typ lékařské zprávy.

**Tabulka 5-3: struktura typů lékařských zpráv [MED\_RecordType]**

Název sloupce	Typ	Pozn.	Popis
RTY_CID	varchar(3)	PK	Kód typu lékařské zprávy
RTY_Name	nvarchar(50)		Název typu lékařské zprávy

Následující tabulka obsahuje přehled výchozích typů lékařských zpráv.

**Tabulka 5-4: výchozí typy lékařských zpráv**

RTY_CID	RTY_Name
AMB	Ambulantní vyšetření
ANM	Anamnéza
LAB	Laboratorní vyšetření
HOS	Hospitalizace
PRE	Výdej léků
DIA	Diagnóza
URG	Urgentní informace
INC	Očkování

Struktura XML fragmentu s obsahem lékařské zprávy ve vztahu k typu lékařské zprávy je odpovídající struktuře standardu DS 04.xx.xx.

#### 5.2.2.2.2 Stavy lékařské zprávy

Pomocí stavů lékařských zpráv jsou v datovém modelu situace popisující:

- Vnitřní stav zprávy (např. „rozpracovaná“, definováno externím zdrojem zprávy)
- Vývoj lékařské zprávy v čase (novější vs. starší verze zprávy)
- Dostupnost zprávy (žádost o výmaz, omezení zobrazení nebo „odstranění“)

S ohledem na potřebu definovat více skupin stavů lékařských zpráv, slouží pro jejich evidenci několik objektů.

**Tabulka 5-5: přehled tabulek s definicemi stavů lékařských zpráv**

Název tabulky	Popis
[MRE_State]	Stavy lékařských zpráv definované v rámci vybrané skupiny stavů
[MRE_StateType]	Definice skupin stavů lékařských zpráv
[MRE_RecordState]	Stavy konkrétních lékařských zpráv, nejvýše jeden stav v rámci skupiny a zprávy

Definice skupin jednotlivých stavů uvedené v předchozích odstavcích jsou uloženy v tabulce [MRE\_StateType]. Každá skupina vymezuje množinu souvisejících stavů lékařské zprávy.

**Tabulka 5-6: struktura skupin stavů lékařských zpráv [MRE\_StateType]**

Název sloupce	Typ	Pozn.	Popis
RST_CID	varchar(3)	PK	Kód skupiny stavů lékařské zprávy
RST_Name	nvarchar(50)		Název skupiny stavů lékařské zprávy

**Tabulka 5-7: výchozí skupiny stavů lékařských zpráv**

RST_CID	RST_Name
ACC	Stav omezení přístupu
INT	Interní stav zprávy
STA	Zobrazení zprávy
SYS	Systémový stav zprávy

Přehled dostupných stavů lékařské zprávy je uložen v tabulce [MRE\_State].

**Tabulka 5-8: struktura definice stavů lek. zpráv v rámci skupin [MRE\_State]**

Název sloupce	Typ	Pozn.	Popis
RES_ID	int	PK	Id definice stavu lékařské zprávy (bez identity)
RES_Type_FK	varchar(3)	PK, FK	Kód skupiny stavů zprávy v [MRE_StateType].[RST_CID]
RES_Name	nvarchar(50)		Název definice stavu lékařské zprávy

Vlastní stavy tykající se konkrétní lékařské zprávy jsou evidovány v tabulce [MRE\_RecordState]. Pro každou skupinu stavů může mít záznam definován nejvýše jeden platný stav, přičemž hodnoty stavů lze na rozdíl od obsahu zprávy v průběhu existence záznamu měnit.

**Tabulka 5-9: struktura stavů lékařské zprávy [MRE\_RecordState]**

Název sloupce	Typ	Pozn.	Popis
RCS_CrDate	datetime		Datum založení záznamu
RCS_CrUser	bigint		Autor záznamu (Id uživatele, není cizí klíč)
RCS_Record_FK	bigint	PK	Id lékařské zprávy v [MRE_Record].[REC_ID]
RCS_State_FK	int	PK, FK	Id definice stavu zprávy v [MRE_State].[RES_ID]
RSS_Type_FK	varchar(3)	PK, FK	Kód skupiny stavů zprávy v [MRE_StateType].[RST_CID]

#### 5.2.2.2.3 Lékařská zpráva

Lékařská zpráva, lze taky zvolit označení „zápis“, je hlavní položkou EZD. Informace lékařské zprávy jsou uloženy v tabulce [MRE\_Record]. Typ zprávy má bezprostřední vliv nejen na strukturu obsahu dané zprávy, ale také na formu přiložených informací jako jsou léky nebo soubory a dále vazeb na další lékařské zprávy. Vlastní obsah lékařské zprávy je uložen do sloupce [REC\_Content], kde je strukturován do formy XML dokumentu. Struktura XML

dokumentu se liší v závislosti na zvoleném typu zprávy, například jiný formát má ambulantní vyšetření a očkování (dle standardu DS 04.xx.xx).

Název sloupce	Typ	Pozn.	Popis
REC_CrDate	datetime		Datum založení záznamu
REC_CrUser	bigint		Autor záznamu (Id uživatele, není cizí klíč)
REC_ID	bigint	PK	Id lékařské zprávy
REC_Type_FK	varchar(3)	FK	Kód typu lékařské zprávy v [MRE_RecordType].[RTY_CID]
REC_Parent_FK	bigint	FK	Nadřazená zpráva (cizí klíč na [MRE_Record][REC_ID])
REC_SourceData_FK	bigint	FK	Id zdrojových dat v [DAT_Import].[DIM_ID]
REC_Patient_FK	bigint	FK	Id entity pacienta v [MED_Patient].[PAT_ID]
REC_Facility_FK	bigint	FK	Id entity ZZ v [MED_Facility].[FAD_ID]
REC_Doctor_FK	bigint	FK	Id entity ZP v [MED_Doctor].[DRE_ID]
REC_Expertise_FK	bigint	FK	Id lékařské odbornosti v [MED_Expertise].[MEX_ID]
REC_DatePublished	datetime		Datum vydání lékařské zprávy
REC_Hash	varchar(50)		Hash vypočítaný z hodnot daného řádku
REC_Content	xml		Obsah lékařské zprávy ve formátu XML

Struktura XML dokumentu s obsahem lékařské zprávy uložené ve sloupci [REC\_Content] je pro každý typ zprávy definována zvlášť.

Primárním zdrojem dat pro tabulku [MRE\_Record] je automatické rozhraní pro import lékařských zpráv.

Z důvodů snadnější identifikace existujícího záznamu vůči datům získaným pomocí automatického rozhraní je v této tabulce obsažen sloupec [REC\_Hash]. Tento sloupec obsahuje hash hodnotu vypočítanou z vybraných údajů dané lékařské zprávy. Porovnáním této hodnoty s hodnotou získanou ze zdrojových dat lze pak snáze identifikovat, zda aktuálně zpracovávaná lékařská zpráva již v databázi existuje či nikoliv.

#### 5.2.2.2.4 Přiložené soubory lékařské zprávy

Ke každé lékařské zprávě lze připojit přílohy ve formě binárních souborů. Přiložené soubory jsou evidovány v tabulce [MRE\_RecordFile] přičemž u každého souboru je rozlišen typ souboru. Typ souboru vyjadřuje, o jaký druh přílohy lékařské zprávy se jedná z pohledu EZD.

Typy přiložených souborů lékařské zprávy jsou uloženy v tabulce [MRE\_RecordFileType].

**Tabulka 5-10: struktura typů přiložených souborů LZ [MRE\_RecordFileType]**

Název sloupce	Typ	Pozn.	Popis
RFT_CID	varchar(3)	PK	Id typu přiloženého souboru
RFT_Name	nvarchar(50)		Název typu přiloženého souboru



**Tabulka 5-11: výchozí typy příložených souborů LZ**

RFT_CID	RFT_Name
FIN	Závěr
LAB	Stará lab. Zpráva
ATT	Příloha

Název sloupce	Typ	Pozn.	Popis
RFL_CrDate	datetime		Datum založení záznamu
RFL_CrUser	bigint		Autor záznamu (Id uživatele, není cizí klíč)
RFL_ID	bigint	PK	Id příloženého souboru
RFL_Type_FK	varchar(3)	FK	Kód typu souboru v [MRE_RecordFileType].[RFT_CID]
RFL_Record_FK	bigint	FK	Id lékařské zprávy v [MRE_Record][REC_ID]
RFL_FileName	nvarchar(50)		Fyzický název souboru
RFL_Data	image		Binární data souboru

#### 5.2.2.2.5 Léky

Součástí lékařské zprávy mohou být i předepsané / vydané léky. Údaje o lécích jsou uloženy v tabulce [MRE\_RecordMedication]. Datový model DASTA umožňuje připojit léky k libovolné lékařské zprávě, v praxi se nicméně léky přikládají pouze k následujícím typům zpráv:

- Ambulantní vyšetření (tzv. předpis léků)
- Výdej léků

Název sloupce	Typ	Pozn.	Popis
RMD_CrDate	datetime		Datum založení záznamu
RMD_CrUser	bigint		Autor záznamu (Id uživatele, není cizí klíč)
RMD_ID	bigint	PK	Id předepsaného léku
RMD_Record_FK	bigint	FK	Id lékařské zprávy v [MRE_Record][REC_ID]
RMD_Name	nvarchar(250)		Název léku
RMD_Code	nvarchar(6)		Kód léku (např. v rámci NČLP)
RMD_MedicineFrom	datetime		Datum podávání léku od
RMD_MedicineTo	datetime		Datum podávání léku do
RMD_Package	float		Údaje o balení
RMD_Dosage	nvarchar(50)		Dávkování léku

## 6 Ukázka aplikace EZD

Jako ukázkovou aplikaci v rámci elektronické zdravotní dokumentace jsem si zvolil detailní návrh implementace webové brány pro automatické přijímání elektronických zpráv pro potřeby EZD. Těmito zprávami mohou být například následující:

- Zdravotní záznamy ze ZIS (zdravotních informačních systémů)
- Kontrola Klientů/Zdravotních pracovníků/Zdravotních zařízení v EZD
- Registrace klientu od autorizovaných systémů (např. Česká Pošta)
- Poskytování/předávání dat a informací z/do třetích systému pomocí autority EZD (federovaná autentizace)

Jako název brány jsem zvolil název B2C brána (business to customer), i když by se spíše hodil název B2B (business to business), ovšem tento název je hojně využíván velkými integračními řešeními pro robustní výměny dat mezi dvěma subjekty. Toto řešení počítá s naopak s větším množstvím různých systémů konzumujících nabízené služby.

B2C rozhraní pro klienty EZD je vystavěno na standardech webových služeb a jejich rozšiřujících specifikací WS-Security, WS-Trust, WS-Policy.

Výměna požadavků pomocí B2C brány je řešena jako asynchronní, tato forma komunikace byla zvolena, protože na některé druhy požadavků nelze odpovědět synchronně v přijatelném čase pro dotazující systém. B2C brána zpřístupňuje klientům asynchronní služby platformy EZD. V rámci tohoto rozhraní je definováno několik webových služeb pro odesílání požadavků do EZD. Dále je definována webová služba pro komunikaci s datovou schránkou klienta – do ní jsou doručovány odpovědi na jednotlivé požadavky, případně ostatní zprávy adresované klientům.

### 6.1 Architektura brány B2C

#### 6.1.1 Zabezpečení a autentizace

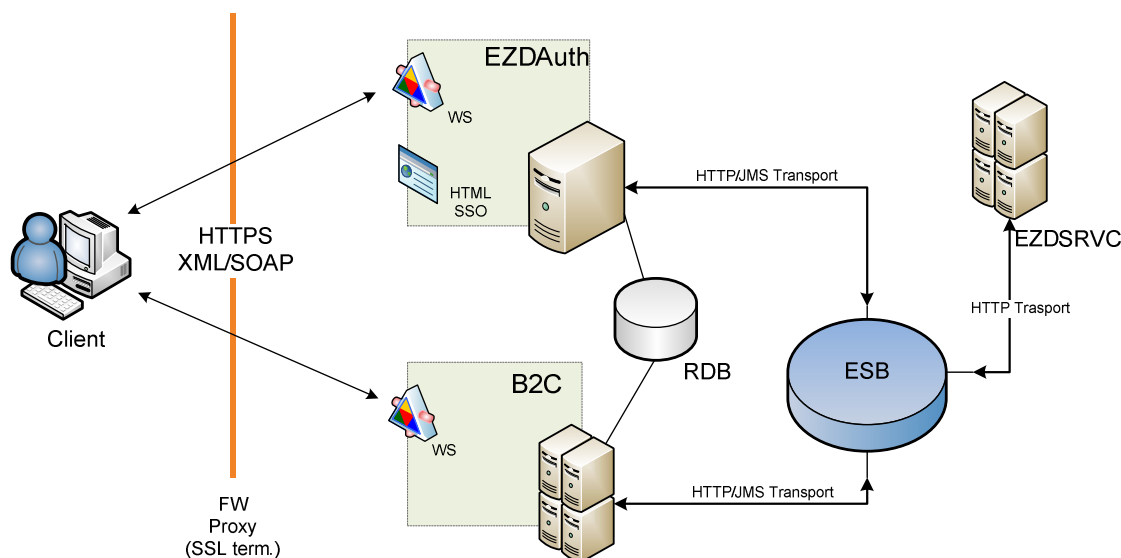
B2C rozhraní je přístupné především pro zdravotnické pracovníky resp. pro jejich zdravotní SW. Systém bude připraven i pro přístup subjektů mimo EZD. Autentizace probíhá za pomoci externího systému EZDAuth, kdy se jedná o webovou službu.

#### 6.1.2 Zabezpečení komunikace

Komunikace zdravotních SW s B2C kanálem bude probíhat pomocí zabezpečeného komunikačního kanálu SSL. V rámci SSL bude použita serverová autentizace, klientská autentizace bude případně použita pouze při autentizaci vůči EZDAuth. Volitelně lze doplnit zabezpečení na úrovni jednotlivých požadavků webových služeb dle standardu WS-Security.

### 6.1.3 Autentizace uživatelů a požadavků

Autentizace požadavků vůči webovým službám bude zabezpečena dle standardu WS-Security Authentication, konkrétně pomocí profilu SAML Token Profile. Autoritou vydávající SAML tokeny bude systém EZDAuth.



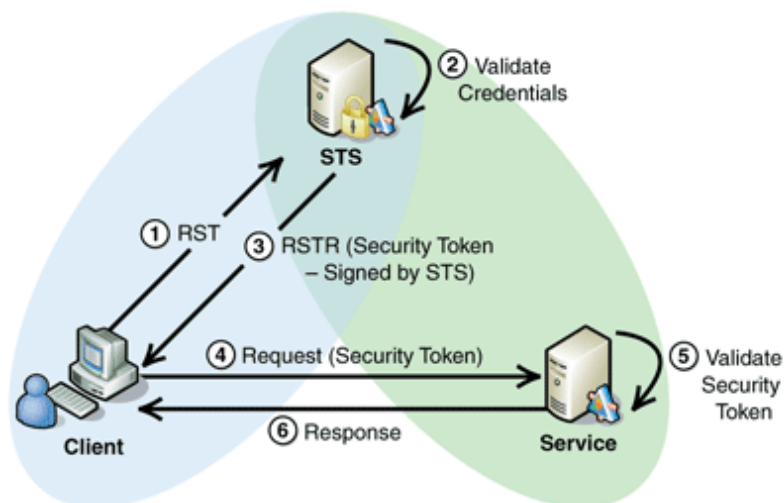
Obr. 6-1: obecná architektura systému

## 6.2 Specifikace služby

### 6.2.1 Autentizace požadavků - EZDAuth

Všechny požadavky na webové služby B2C rozhraní jsou autentizovány dle standardu WS-Security (SAML Token Profile). Autentizační tokeny jsou vystavovány STS (Security Token Service) externí službou EZDAuth dle standardu WS-Trust.

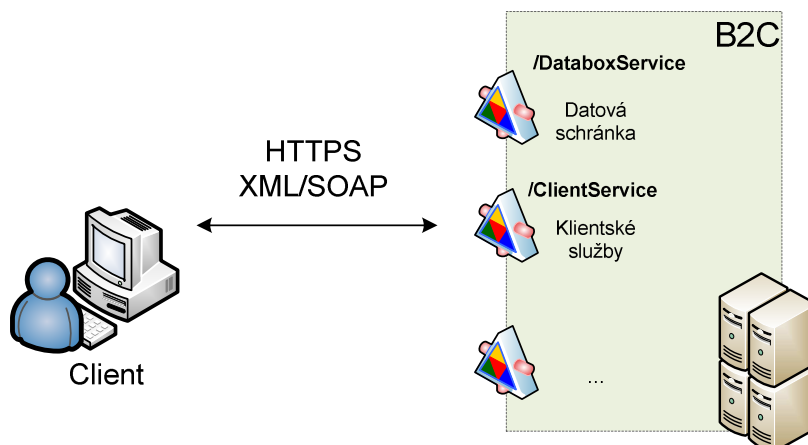
Na následujícím obrázku je schematicky znázorněn proces autentizace uživatele, získání autentizačního tokenu a následně volání služby. Klienti se vůči STS serveru autentizují pomocí svého identifikačního čísla v rámci systému EZD a pomocí PINu. Tento autentizační token je vystaven s určitou časovou platností (1-2 hodiny) a je součástí zpráv pro komunikaci s B2C bránou. Toto zejména usnadňuje celkovou komunikaci, kdy se klient přistupující na bránu B2C nemusí pro zaslání jednotlivých zpráv autentizovat vůči systému EZD, ale postačí mu pouze autentizační token s omezenou časovou platností. Toto vede i ke snížení zátěže kladené na systém jako takový.



Obr. 6-2: autentizace požadavků služeb

## 6.2.2 Popis brány

Brána B2C je rozdělena na dvě samostatné části, kde „ClientService“ je soubor webových služeb sloužících pro komunikaci se systémem EZD. Dále je definována webová služba „DataboxService“ pro komunikaci s datovou schránkou klienta, do které jsou doručovány odpovědi na jednotlivé požadavky, případně ostatní zprávy adresované klientům.



Obr. 6-3: struktura B2C rozhraní

## 6.2.3 Datová schránka

Do datové schránky jsou doručovány odpovědi na odeslané zprávy. Každý klient má datovou schránku zřízenou trvale. Zprávy jsou do ní doručeny i po výpadku komunikace. Datová schránka je přístupná pomocí metod webové služby /DataboxService.

Zprávy jsou v rámci datové schránky řazeny ve frontě. Z fronty lze vybrat buď první zprávu, nebo konkrétní zprávu dle korelačního identifikátoru. Výběr zprávy je třeba potvrdit do krátkého časového intervalu (např. 5 min.) po vyzvednutí. V případě, že se toto nestane, je zpráva zařazena zpět do schránky na konec fronty.

Nad datovou schránkou lze provádět tyto operace:

- Výběr odpovědi dle korelačního identifikátoru
- Výběr první zprávy ve schránce
- Potvrzení přijetí zprávy

### 6.2.3.1 Výběr odpovědi dle korelačního identifikátoru

Tato metoda umožňuje výběr zprávy z datové schránky pomocí identifikátoru, který byl klientskému systému navrácen při odeslání zprávy do B2C rozhraní. Tento korelační identifikátor slouží pro vybrání určité specifické zprávy.

Název metody: `ReceiveById`

**Tabulka 6-1: vstupní parametry metody [ReceiveById]**

Název parametru	Typ	Povinné	Popis
Corid	string	ano	Korelační identifikátor zprávy, pro kterou chceme načíst odpověď.

**Tabulka 6-2: výstupní parametry metody [ReceiveById]**

Název parametru	Typ	Popis
Msgid	string	Identifikátor načtené zprávy. V případě, že odpovídající zpráva ve schránce neexistuje tak nil.
Corid	string	Korelační identifikátor zprávy. Shodný s identifikátorem Corid v požadavku.
Type	string	Identifikace typu zprávy – v současnosti může obsahovat pouze hodnotu client (klientské služby)
Data	Xml	Vlastní Xml zpráva s odpovědí.

V případě, že ve schránce odpověď na danou zprávu neexistuje, je klientskému systému navrácen identifikátor zprávy `Msgid = nil`. Hodnoty v ostatních parametrech nejsou v tomto případě platné.

Úspěšný výběr zprávy je nutné potvrdit voláním metody `Confirm`, které je předán parametr `Msgid`.

### 6.2.3.2 Výběr první zprávy ve schránce

Pomocí této metody je vyzvednuta první zpráva ve schránce. V případě, že žádná zpráva ve schránce neexistuje, je v parametru `Msgid` navracena hodnota `nil`. Tato metoda nepotřebuje a nemá žádné vstupní parametry.

Název metody: `Receive`

**Tabulka 6-3: výstupní parametry metody [Receive]**

Název parametru	Typ	Popis
Msgid	string	Identifikátor načtené zprávy. V případě, že odpovídající zpráva ve schránce neexistuje tak nil.
Corid	string	Korelační identifikátor zprávy.
Ttype	string	Identifikace typu zprávy – v současnosti může obsahovat pouze hodnotu client (klientské služby)
Data	Xml	Vlastní Xml zpráva s odpovědí.

Výstupní parametry metody jsou ve skrze identické jako v případě metody `ReceiveById` rozdíl je pouze ve způsobu volání.

### 6.2.3.3 Potvrzení přijetí zprávy

Potvrzení přijetí zpráv klientem. V případě, že není přijetí zprávy potvrzeno do krátkého časového intervalu (např. 5 min.) od volání metod `Receive` a `ReceiveById`, je zpráva navrácena zpět do datové schránky a to na konec fronty.

Název metody: `Confirm`

**Tabulka 6-4: Vstupní parametry metody [Confirm]**

Název parametru	Typ	Povinné	Popis
msgid	string	ano	Identifikátor zprávy, jejíž přijetí potvrzujeme.

Tato metoda nevrací žádné výstupní parametry.

## 6.2.4 Klientské služby

Klientské služby EZD jsou do B2C rozhraní zpřístupněny pomocí metod webové služby `/ClientService`. Všechny požadavky na klientské služby jsou předávány pomocí požadavků ve formátu XML. Odpovědi na tyto požadavky jsou doručeny do datové schránky jako zprávy s typem `client`.

U všech sad XML dokumentů je vyžadováno, aby byly ve standardu XML Signature opatřeny autentizačními prvky, např. elektronickým podpisem klienta vytvořeným na základě kvalifikovaného certifikátu nebo přihlašovacími údaji.

### 6.2.4.1 Odeslání požadavku

Název metody: `Send`

**Tabulka 6-5: vstupní parametry metody [Send]**

Název parametru	Typ	Povinné	Popis
Data	Xml	ano	XML dokument s požadavkem

**Tabulka 6-6: výstupní parametry metody**

Název parametru	Typ	Popis
Msgid	<code>string</code>	Identifikátor přidělený zprávě. Lze se pomocí něj dotazovat na výsledek zpracování.

## 7 Závěr

### 7.1 Přínos práce

Hlavní obecný přínos této diplomové práce lze rozdělit na dvě části. První je obecný popis EZD jako takové, důvod tvorby takového systému, jeho použití a přínosy, dále je to rešerše na téma datových standardů, jejich historii, popis a způsob práce s nimi. Výsledkem této rešerše je krátký závěr, v jehož výsledku jsem se rozhodl použít datový standard DS. Použití tohoto standardu bylo podmíněno, jak detailní znalostí tohoto standardu tak i tím, že pro zvolenou problematiku EZD plně vyhovuje z pohledu zpracování a uložení zdravotní zprávy.

Pro standard HL7 hovoří je komplexnost popisu daného prostředí, kdy nabízí nástroje na popis většiny možných situací a jejich vývoj v čase. Ovšem jeho rozsah je zároveň i překážkou pro práci s tímto standardem, kdy jakákoliv implementace je velmi náročná. I tak bych se ovšem v budoucnu s tímto standardem rád setkal.

Součástí této rešeršní částí diplomové práce je i část věnující se problematice webových služeb, jako nástroji pro komunikaci mezi systémem EZD a dalšími systémy ve zdravotnictví. Použití webových služeb jako komunikačního rozhraní EZD z mého pohledu nabízí ucelenou formu komunikace, pomocí které se dá realizovat jak jednoduchý adaptér pro předání dat systému, tak i možnost realizovat dlouhodobé asynchronní volání jiných služeb.

Druhou částí je moje vlastní analýza takového systému, která je rozpracována v kapitole 5 Analýza systému EZD. Účelem této analýzy je prvotní a základní náhled na zpracovávanou problematiku EZD z pohledu klientsky orientovaného přístupu, kdy klient je majitelem zdravotních záznamů zde uvedených. Soustředím se na způsoby použití systému EZD, přístupu k informacím a role jednotlivých uživatelů. Výstupem je popis činností, které by mělo EZD vykonávat. Identifikuji zde jednotlivé uživatele, jejich role a práva v systému a způsoby práce se zdravotními a jinými záznamy. V práci je uveden pouze výtah z celkové zpracované analýzy, celá dokumentace týkající se Use Case analýzy je k nalezení na přiloženém CD.

Dále se věnuji základnímu popisu jak postupovat při tvorbě návrhu databázové architektury úložiště s důrazem na jednoduchost a zároveň obecnost, pro jeho snadnou rozšiřitelnost. Zde je ukázáno, že použití XML nemusí být nutně nejvhodnější formou uložení dat, jako výhodnější se jevila možnost použití kombinace databáze a XML, kdy v XML je uložena informace o samotné zdravotní zprávě a databáze slouží k identifikaci vazeb okolních objektů vztažených k samotné zprávě (zdravotnický pracovník, pacient, další lékařské zprávy).

Jako poslední je uvedena dokumentace názorné implementace rozhraní pro snadnou komunikaci externích systému se systémem EZD, za použití webových služeb.



## 7.2 Co v práci není

V práci jsem se nezabýval ochranou dat z pohledu zákona o ochraně osobních údajů. V části analýzy je ovšem vývoj takového systému modelován s ohledem na dodržování ochrany osobní údajů.

V práci také není kompletní datový model, pro úplný model by bylo třeba doplnit entity týkající se přístupů, uživatelských účtů, jejich vazby na zprávy, odbornosti. Dále pak geografických entit, historie a logování, kontaktní údajů a podobně.

Práci jako takovou by bylo možné dále rozšířit o návrh aplikačního modelu (kapitola 5 Analýza systému EZD). Tento model bych stavěl s důrazem na rozdělení na oddělení jednotlivých aplikačních vrstev, konkrétně na datovou, business a uživatelskou vrstvu. Kde datová vrstva by definovala datové objekty a sloužila pro komunikaci s databází. Business vrstva by obsahovala aplikační jádro, framework s definicí služeb. Uživatelská vrstva by obsahovala definici rozhraní pro uživatele, tvořeného pomocí návrhového vzoru MVC<sup>33</sup> nebo MVP<sup>34</sup>.

## 7.3 Budoucnost

Tak jak je v celé práci prezentován systém Elektronické Zdravotní Dokumentace, jako systém, kde majitelem zdravotních dat je pacient, neodpovídá v tuto chvíli realitě. Osobní údaje i informace o zdravotním stavu jsou vlastnictvím pacienta, avšak zdravotnická dokumentace je ve vlastnictví zdravotnického zařízení. Pacient má pouze nárok na kopii své zdravotní dokumentace. Pokud by v budoucnu nedošlo k žádné změně, tak i v tomto případě by měl zde popsán způsob šanci na úspěch. V případě kdy by se dále zdravotní dokumentace vedla stávajícím způsobem, stačilo by tento systém prohlásit za „Kopii zdravotní dokumentace v elektronické podobě“ a všechny jeho zde popsané přínosy by zůstaly zachovány. Další velký tlak na elektronický vedenou zdravotní dokumentaci (interoperabilní) je vyvíjen ze strany Evropské Unie, jak jednotlivými projekty jako jsou EPSOS , DebugIT a jiné, tak i celkovým přístupem k této problematice. Podobný trend lze vysledovat i ve Spojených státech amerických, kdy prezident Obama prohlásil, že do dvou let bude mít každý americký občan elektronicky vedenou zdravotní dokumentaci. Ovšem v tuto chvíli jsem nenalezl model či popis jak se tento systém má chovat.

---

<sup>33</sup> MVC – Model View Controller

<sup>34</sup> MVP – Model View Presenter

## 8 Seznam literatury

1. **Němec, Lukáš.** *Elektronická správa dokumentace.* Praha : Bakalářská práce - Česká zemědělská univerzita v Praze, 2007.
2. **Eysenbach, G.** What is e-health? *Journal of Medical Internet Research.* [Online] 2001. <http://www.jmir.org/2001/2/e20/>.
3. *356: Communication from the Commission to the Council, the European Parliament, the European Economic and Social Committee and the Committee of the Regions: eHealth - making healthcare better for European citizens: An action plan for a European eHealth. Communities, Commission of the European.* Brussels : autor neznámý, 2004.
4. *Electronic business in the health and social services sector – Sector Impact Study No. 10-I (draft). The European e-business W@tch 2003/4, European Commission, Enterprise Directorate General. Stroetmann, Stroetmann K.A. and V.N.* Brussels/Bonn : autor neznámý, 2004.
5. **ČLS JEP, MZČR.** Datový standard MZČR. *Datový standard Ministerstva Zdravotnictví České Republiky.* [Online] 2006. [Citace: 19. 02 2009.] <http://ciselniky.dasta.mzcr.cz/>.
6. **Kašpar, Václav.** Nástroj pro popis libovolného zdravotnického informačního... <http://fel.cvut.cz>. [Online] 2007. [Citace: 11. 01 2009.]
7. **Wikipedia.** Wikipedia. *Wikipedia EN.* [Online] 2008. [Citace: 9. 1 2009.] <http://en.wikipedia.org/wiki/HL7>.
8. **Hanzlíček, Petr.** HL7 Česká Republika. *HL7 Česká Republika.* [Online] 2008. [Citace: 8. 1 2009.] [http://www.hl7.cz/file/53/HL7v3\\_medsoft2005.pdf](http://www.hl7.cz/file/53/HL7v3_medsoft2005.pdf).
9. **Robert H. Dolin, MD, Liora Alschuler, Sandy Boyer, BSP, Calvin Beebe, Fred M. Behlen, PHD, Paul V. Biron, Amnon Shabo (shvo), PHD.** <http://www.jamia.org/cgi/reprint/13/1/30>. *HL7 Clinical Document Architecture.* [Online] HL 7, 2008. [Citace: 9. 1 2009.]
10. **Kosa, Vít.** HL7 Česká republika. *HL7 Česká republika.* [Online] 1. 12 2008. [Citace: 8. 1 2009.] <http://www.hl7.cz/file/14/StrucnyPopisV3.pdf>.
11. **HL7.** Významový slovník terminologie používané standardy HL7. <http://www.hl7.cz>. [Online] HL7 Česká Republika, 2007. [Citace: 9. 1 2009.]
12. **Smith, Barry.** <http://hl7-watch.blogspot.com/>. *HL7 Watch.* [Online] 2008. [Citace: 15. 01 2009.]
13. **W3C, IETF.** HTTP. [Online] 2009. [Citace: 26. 2 2009.] <http://www.ietf.org/internet-drafts/draft-ietf-httpbis-p1-messaging-05.txt>.

14. **W3C**. W3C - SOAP. [Online] 2001. [Citace: 2. 24 2009.] <http://www.w3.org/TR/soap12-part0/>.

15. —. W3C - WSDL definition. [Online] 2001. [Citace: 25. 2 2009.] <http://www.w3.org/TR/wsdl>.

16. **Journal of Medical Internet research**. Medicine 2.0 Theme Issue: Web 2.0 in Health, Health Care, Medicine, and Biomedical Research. *Journal of Medical Internet research*. [Online] 2008. [Citace: 4. 1 2009.] <http://www.jmir.org/2008/3>.

## 8.1 Seznam obrázků

Obr. 4-1: DS verze 01.01. ....	13
Obr. 4-2: DASTA - blokové schéma.....	18
Obr. 4-3: kostra Referenčního Informačního Modelu (zdroj HL7 NPO).....	23
Obr. 4-4: příklad reprezentace zdravotního výkonu v RIM HL7 .....	23
Obr. 4-5: hierarchická struktura třídy Entita (zdroj HL7 NPO) .....	24
Obr. 4-6: hierarchická struktura třídy Role (zdroj HL7 NPO).....	25
Obr. 4-7: hierarchická struktura třídy Participace a Akce (zdroj HL7 NPO).....	26
Obr. 4-8: schéma komunikace pomocí WS.....	31
Obr. 4-9: struktura WSDL .....	33
Obr. 4-10: struktura SOA architektury.....	36
Obr. 4-11: ukázka WS orchestrace procesu v BPEL .....	37
Obr. 5-1: přístup klienta do webového rozhraní EZD .....	40
Obr. 5-2: aktivity klienta v systému EZD .....	41
Obr. 5-3: aktivity zdravotního pracovníka v systému EZD.....	42
Obr. 5-4: přístup zdravot. zařízení/pracovníka pomocí B2C brány .....	43
Obr. 5-5: zápis a nahlížení nad zdravotním záznamem.....	44
Obr. 5-6: schéma lékařských zpráv a závislých záznamů (ROE).....	48
Obr. 5-7: schéma lékařských odborností .....	49
Obr. 5-8: schéma lékařské zprávy .....	51
Obr. 6-1: obecná architektura systému .....	57
Obr. 6-2: autentizace požadavků služeb.....	58
Obr. 6-3: struktura B2C rozhraní.....	58

## 8.2 Seznam příkladů

Příklad 4-1: příklad datového souboru DS 04.02.02 - klinická událost.....	18
Příklad 4-2: příklad NČLP číselníku .....	19

## 8.3 Seznam tabulek

Tabulka 5-1: struktura skupin zdrav. odborností [MED_ExpertiseGroup] .....	49
Tabulka 5-2: struktura odborností [MED_Expertise] .....	50
Tabulka 5-3: struktura typů lékařských zpráv [MED_RecordType] .....	52
Tabulka 5-4: výchozí typy lékařských zpráv.....	52

Tabulka 5-5: přehled tabulek s definicemi stavů lékařských zpráv.....	52
Tabulka 5-6: struktura skupin stavů lékařských zpráv [MRE_StateType] .....	53
Tabulka 5-7: výchozí skupiny stavů lékařských zpráv.....	53
Tabulka 5-8: struktura definice stavů lek. zpráv v rámci skupin [MRE_State]...	53
Tabulka 5-9: struktura stavů lékařské zprávy [MRE_RecordState].....	53
Tabulka 5-10: struktura typů přiložených souborů LZ [MRE_RecordFileType] .	54
Tabulka 5-11: výchozí typy přiložených souborů LZ .....	55
Tabulka 6-1: vstupní parametry metody [ReceiveByID] .....	59
Tabulka 6-2: výstupní parametry metody [ReceiveByID] .....	59
Tabulka 6-3: výstupní parametry metody [Receive].....	60
Tabulka 6-4: Vstupní parametry metody [Confirm].....	60
Tabulka 6-5: vstupní parametry metody [Send] .....	60
Tabulka 6-6: výstupní parametry metody.....	61

## 9 Přílohy

### 9.1 Dokumentace Use Case

Přílohou diplomové práce je dokumentace Use Case vytvořené v rámci praktické části. Ve formě exportu z Enterprise Architectu 7 jsou zdrojové kódy HTML dokumentace přiloženy v elektronické podobě ve formě CD nosiče.

### 9.2 Seznam zkratek

<b>Zkratka</b>	<b>Popis</b>
HL7	<i>Health Level Seven</i>
DS/DASTA	<i>Datový standard Ministerstva zdravotnictví ČR</i>
WS	<i>Web services – webové služby</i>
XML	<i>Extensible Markup Language</i>
SQL	<i>Strucured Query Language</i>
UML	<i>Unified Modeling Language</i>
SOA	<i>Service Oriented Architecture</i>
SOAP	<i>Simple Object Access Protocol</i>
SNOMED CT	<i>Systematized Nomenclature of Medicine - Clinical Terms</i>
CEN	<i>European Comittee for Standardization</i>
XSLT	<i>eXtensible Stylesheet Language Transformations</i>
LČLP	<i>lokální číselník laboratorních položek</i>
ANSI	<i>Americký Národní institut pro normalizaci</i>
SDO	<i>Standard Developing Oragnization</i>
ISO	<i>Mezinárodní organizace pro normalizaci</i>
RIM	<i>Referenční Informační Model</i>
ASTM	<i>American Society for Testing and Materials</i>
ACR	<i>American College of Radiology</i>
X12N	<i>Acredited Standards Committee</i>
IEEE	<i>Institute of Electronics and Electrical Engineers</i>
XSD	<i>XML schema definiton</i>
CMETs	<i>Common Message Element types</i>
ITS	<i>Implementation Technology Specification</i>
D-MIM	<i>Domain Message Information Model</i>
R-MIM	<i>Refined Message Information Model</i>
HMDs	<i>Hierarchal Message Definitions</i>
LONIC	<i>Logical Observation Identifiers Names and Codes</i>
WSDL	<i>Web Services Description Language</i>
UML	<i>Unified Modeling Language</i>
MVC	<i>Model View Controler</i>

MVP	<i>Model View Presenter</i>
BPEL	<i>Business Process Execution Language</i>

### 9.3 Seznam pojmů

<b>Pojem</b>	<b>Popis</b>
Klient	Software přistupující ke službě jiného programu nebo počítače (server).
Use Case	Use Case model je grafickým zobrazením části dokumentu specifikace požadavků. Požadavky na informační systém lze rozdělit na nefunkční a funkční.
Relační databáze	databázový systém, který je založen na relačním modelu dat a relační algebře. Data jsou uspořádána do tabulek (relací), nad kterými jsou definovány přípustné operace.
Server	Počítač nebo druh softwarového vybavení zajišťující služby pro ostatní počítače.
Webová služba	Systém sloužící pro komunikaci a výměnu dat mezi různými aplikacemi a platformami. Webová služba je často pouze formou vzdáleného API rozhraní, které slouží ke spuštění procesů nebo služeb v rámci daného serveru.