

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE
Provozně ekonomická fakulta
Katedra informačního inženýrství



**Kvantifikovaný přístup k jakosti informačního zabezpečení
pro podporu evaluace informačních technologií**

Disertační práce

Autor: Ing. Michal Příbrský
Školitel: Doc. Ing. Prokop Toman, CSc.
Obor: Informační management

© 2012

Děkuji svým školitelům Prof. RNDr. Jiřímu Vaníčkovi, CSc., In memoriam, a Doc. Ing. Prokopovi Tomanovi, CSc. za odborné vedení této disertační práce.

Dále děkuji kolegům Ing. Robertovi Perglovi, PhD. a Ing. Martinovi Papíkovi, PhD. za cenné konzultace při vzniku této práce.

Souhrn

Tato disertační práce se zabývá návrhem metody a měr ke kvantifikovanému přístupu k jakosti informačního zabezpečení podniků. V první, rešeršní, části práce jsou shrnuty poznatky o jakosti softwarových produktů. Dále je nastíněn přehled současného stavu mezinárodní normalizace jakosti software, kde zvláštní pozornost je věnována probíhajícímu mezinárodnímu projektu SQaRE, který má za cíl nahradit stávající normy novou konzistentní soustavou. Část práce o jakosti softwarových produktů uzavírá princip a způsob měření jakosti a provádění auditu IS.

Na základě získaných teoretických poznatků z rešeršní části disertační práce a formulovaných předpokladů je navržena metoda pro kvantifikovaný přístup k řízení jakosti software. Navržená metoda je kombinací teoretické a empirické části disertační práce, jejímž výsledkem je model prvků měření kvality. Jde o optimalizaci expertních odhadů a požadavků na jakost software.

Navržená metoda byla ověřena v praxi ve finančním sektoru pro vybraný aplikační software. Ověření navržené metody zahrnuje analýzu obdržených výsledků a způsob výběru prvků do modelu. V závěrečné části práce je navržen výsledný model prvků měření kvality pro hodnocení jakosti konkrétního informačního systému.

Klíčová slova

Jakost, norma, prvek měření kvality, hodnocení jakosti, informační systém, audit informačního systému.

Summary

This Doctoral thesis deals with design of the method and measures for quantified approach to quality management of selected software used within companies. In the first research part of the paper there are summarized the findings about the quality of software products. Furthermore there is also outlined an overview of the current state of international standardization of quality software, where special attention is dedicated to the ongoing international project SQUARE, which has goal to replace existing standards with the consistent new system. Part of the thesis about the quality of software products closed by principle and method of the quality measurement and performing of IS audit.

Based on the theoretical knowledge of the research part of the thesis and formulated assumptions the method is designed for a quantified approach to quality software management. The proposed method is a combination of theoretical and empirical part of the thesis, which results is the quality measure elements model. This is the optimization of expert estimates and software quality requirements.

The proposed method was verified practically in the financial sector for the selected application software. Verification of the proposed method includes an analysis of the results obtained and the process for elements selecting into the model. In the final part of the thesis there is designed resulting model of the quality measure elements for the quality evaluation of a specific information system.

Key words

Quality, Standard, Quality Measure Element, Quality Evaluation, Information System, Information System Audit.

Obsah

OBSAH	5
1 ÚVOD	8
1.1 CÍLE DISERTAČNÍ PRÁCE	10
1.2 METODICKÝ POSTUP	11
1.3 POUŽITÁ TERMINOLOGIE.....	13
1.4 STRUKTURA DISERTAČNÍ PRÁCE	13
1.5 OMEZENÍ A LIMITY PRÁCE.....	14
1.6 SOUČASNÝ STAV ŘEŠENÍ DANÉ PROBLEMATIKY	15
1.7 SOUVISEJÍCÍ TÉMATA, KTERÁ NEJSOU OBSAHEM PRÁCE.....	17
2 JAKOST INFORMAČNÍHO SYSTÉMU	18
2.1 JAKOST PRODUKTU A JAKOST PROCESU.....	20
2.2 JAKOST ETAPY ŽIVOTNÍHO CYKLU	21
2.3 RŮZNÉ POHLEDY NA JAKOST.....	22
3 MEZINÁRODNÍ NORMALIZACE JAKOSTI SOFTWARE	25
3.1 SOUČASNÝ STAV	26
3.1.1 ISO/IEC 12119	27
3.1.2 ISO/IEC 9126	28
3.1.3 ISO/IEC 14598	29
3.1.4 ISO/IEC 15939.....	30
3.2 PERSPEKTIVY MEZINÁRODNÍCH NOREM	30
3.2.1 Model SQuaRE.....	30
3.2.2 ISO/IEC 2500n - Quality Management Division.....	34
3.2.3 ISO/IEC 2501n – Quality Model Division.....	36
3.2.4 ISO/IEC 2502n – Quality Measurement Division.....	37
3.2.5 ISO/IEC 2503n – Quality Requirements Division	40
3.2.6 ISO/IEC 2504n – Quality Evaluation Division.....	41
3.2.7 ISO/IEC 25050 – 25099 SQuaRE Extension Division.....	42
3.2.8 Nedostatky modelu SQuaRE.....	43
3.3 CHARAKTERISTIKY JAKOSTI.....	44
3.3.1 Jakost užití.....	45
3.3.2 Kvalita softwarového produktu.....	46
3.4 ATRIBUTY JAKOSTI	48
4 MĚŘENÍ JAKOSTI INFORMAČNÍCH SYSTÉMŮ	51
4.1 VLASTNOSTI SOFTWARE.....	52
4.2 POŽADAVKY NA JAKOST SOFTWARE	54
4.2.1 Kategorizace požadavků na systém	56

4.2.2	Zúčastněné strany a požadavky zúčastněných stran	57
4.2.3	Požadavky zúčastněných stran a požadavky na systém	58
4.3	POSTUP HODNOCENÍ JAKOSTI	59
4.4	MÍRY ATRIBUTŮ	60
4.4.1	Výběr atributů a měř.	62
4.4.2	Vlastnosti atributů a měř.	63
4.5	MĚŘICÍ STUPNICE	65
4.5.1	Absolutní stupnice	65
4.5.2	Poměrová stupnice	65
4.5.3	Intervalová stupnice	66
4.5.4	Ordinální stupnice	66
4.5.5	Nominální stupnice	67
5	AUDIT INFORMAČNÍHO SYSTÉMU	69
5.1	NĚKTERÉ METODIKY AUDITU IS/IT	69
5.2	OBECNÝ POSTUP PROVÁDĚNÍ AUDITU IS/IT	72
5.2.1	Uzavření smlouvy na audit (0.Etapa)	72
5.2.2	Předběžné plánování (1.Etapa)	75
5.2.3	Vytvoření plánu auditu (2.Etapa)	76
5.2.4	Realizace auditu (3.Etapa)	76
5.2.5	Závěr a vydání auditorské zprávy (4.Etapa)	77
5.2.6	Sledování plnění závěrů auditorské zprávy (5.Etapa)	77
6	KONCEPT QUALITY MEASURE ELEMENTS	78
6.1	PREZENTACE MODELU METODY MĚŘENÍ	78
6.2	ZÁKLADNÍ SADA QME	79
6.3	PROJEKTOVÁNÍ PRVKŮ MĚŘENÍ KVALITY (QME)	85
6.3.1	Identifikace QME a cílů	86
6.3.2	Identifikace vlastnosti ke kvantifikaci v souvislosti s QME	87
6.3.3	Definice vlastnosti a sub-vlastností	88
6.3.4	Stavba modelu vlastnosti ke kvantifikaci	88
6.3.5	Přiřazení jednotky měření (vzorec) a typ měřicí stupnice	88
7	NÁVRH METODY PRO KVANTIFIKOVANÝ PŘÍSTUP K ŘÍZENÍ JAKOSTI IS	90
7.1	INFORMAČNÍ ZABEZPEČENÍ PODNIKŮ	90
7.2	VÝCHOZÍ PŘEDPOKLADY	92
7.3	NÁVRH METODY A MĚŘ PRO KVANTIFIKOVANÝ PŘÍSTUP K ŘÍZENÍ JAKOSTI	95
7.4	OVĚŘENÍ METODY V PRAXI	99
7.5	VYHODNOCENÍ VÝSLEDKŮ	113
7.5.1	Vytvoření výsledného modelu	116
7.5.2	Vytvoření výsledného modelu pomocí software MaxAgr	117

8 ZÁVĚR	120
8.1 SHRnutí DISERTAČNÍ PRÁCE.....	120
8.2 PŘÍNOSY DISERTAČNÍ PRÁCE.....	121
8.3 DISKUSE A NÁMĚTY K DALŠÍMU VÝZKUMU.....	123
SEZNAM OBRÁZKŮ	124
SEZNAM TABULEK	124
SEZNAM ZKRATEK	125
LITERATURA A ZDROJE	126
PŘÍLOHA A	129
PŘÍLOHA B	139

1 Úvod

Význam informatických produktů, které jsou používány ve společnostech a ve firmách, a které mají zabezpečit chod i cíle těchto podniků, v posledních desetiletích stále roste. Na informační technologie obecně jsou vynakládány nemalé finanční prostředky, v případě informačních systémů lze hovořit o rádech jednotek až desítek milionů. Dále lze uvést v současnosti používané tvrzení manažerů, že výdaje do informačních technologií trvale rostou, ale tyto investice nepřinášejí očekávaný užitek. Tato situace vede k tomu, že se management společností zajímá o odpovědi na následující otázky:

- návratnost těchto vynaložených prostředků,
- jaká je celková potřeba těchto produktů,
- jak velké jsou celkové výdaje na informační technologie,
- jaká je vytvořená přidaná hodnota pro podnik s těmito technologiemi, atd.

Podobných otázek by bylo jistě možné najít celou řadu. Přesto další prostředky a zdroje budou do informačních systémů – informačních technologií (IS/IT) investovány ve větší míře, než tomu bylo v minulosti. Důvody mohou být: rostoucí konkurence, vyšší požadavky klientů, výkyvy a vznik nových (globálních) trhů, vědecko-technický rozvoj. Dle prof. Molnára [Molnár 2000] by mělo být snahou racionálních subjektů hledání optimálního (tj. vyváženého) poměru mezi užitekem, který získá z IS/IT a výdaji, které musí na získání tohoto užitku vynaložit, ale také mezi časem potřebným na získání tohoto užitku a riziky spojenými s tím, že tohoto očekávaného užitku nedosáhne. Takto, z hlediska subjektu, “vyvážený” systém pak můžeme považovat za efektivní.

V odborné literatuře, např. v monografiích [Vrana, Richta 2005], [Basl 2002], jsou detailně popsány podnikové informační systémy, jejich výběr, řízení takového projektu, implementace i řízení inovace a změn těchto systémů. Rovněž jsou charakterizovány, a dle autorů prokázány [Molnár 2000], [Učeň et al. 2001] efekty a přínosy zavedení informačních systémů do firem. Jakost (kvalita) a hodnocení těchto produktů je velmi podrobně zpracováno prof. Vaníčkem [Vaníček 2004a], stejně jako způsob provádění auditu informačních systémů [Svatá 2011].

Informační technologie se používají ve stále širší škále aplikačních oblastí a jejich správný provoz je často rozhodující pro obchodní úspěch. Vývoj i výběr softwarového produktu vysoké kvality má proto zásadní význam. Komplexní

specifikace a hodnocení jakosti softwarového produktu je klíčovým faktorem zajištění odpovídající kvality. Při analýze dostupných pramenů se ukázalo, že jsou poměrně důkladně zpracovány postupy provádění hodnocení jakosti i auditů informačních systémů, problémem však je převod mezi potřebami a požadavky na jakost a výběr vhodných atributů a jejich měř. Dle Vaníčka [Vaníček 2008c] získat adekvátní množinu atributů a měř, která jednotlivé charakteristiky jakosti dobře vystihuje, není lehký problém. Tato množina prvků měření kvality musí jakost dostatečně dobře pokrývat, nesmí však být příliš rozsáhlá a komplikovaná, aby neodrazovala možné zájemce o hodnocení kvality software. Pokud je kvalita ve firmách vyvíjejících software měřena, způsob obvykle není veřejnou informací.

Prvky měření kvality (QME) jsou řešeny v rámci projektu SQuaRE a je jim věnována zvláštní norma [ISO/IEC 2011a]. Zde je uvedena určitá statická, dle názoru autora značně omezená množina prvků měření kvality a jejich mapování na charakteristiky a podcharakteristiky jakosti. V pramenech není vymezeno, jakým způsobem bylo právě k navrhovaným prvkům dospěno, ani způsob, jakým bylo vytvořeno mapování. Rovněž není uveden důvod počtu těchto prvků v množině. Pro výběr prvků do množiny a následného vytvoření mapování je třeba zohlednit také typ softwarového produktu a prostředí, ve kterém je implementován. V této práci jsou navrženy některé další prvky, které by bylo vhodné do množiny zařadit. Záměrem však není vytvoření pouze další statické množiny, ale navrhnout metodu pro výběr těchto prvků, které budou přizpůsobeny potřebám pro určitý konkrétní produkt používaný v určitém konkrétním prostředí společnosti.

Výše uvedené skutečnosti vedly autora k formulaci předpokladů pro vznik této práce: základní sada prvků měření kvality přes svoji univerzálnost nepokrývá praktické potřeby, a proto je nutné seznam upravit a rozšířit o další měřitelné prvky, a současně podle konkrétních potřeb hodnotitelů je nutné změnit a přepracovat mapování prvků měření kvality na charakteristiky a podcharakteristiky jakosti software.

Tato disertační práce je věnována návrhu metody a měř pro kvantifikovaný přístup k řízení jakosti softwarových produktů. Explicitní vymezení pohledu, který je v práci popsán, autor v žádném z dostupných pramenů nenalezl. Jde o vytvoření metody sběru uživatelských požadavků založených na charakteristikách jakosti. Tento inženýrský problém vykazuje jistou analogii s analýzou informačních systémů: při analýze IS identifikujeme atributy objektů v doméně, zatímco zde identifikujeme

měřitelné atributy vhodné jako vstup pro hodnocení a audit jakosti informačního systému.

Teoretické výsledky výzkumu budou ověřeny v praxi, předpoklad je využití některých informačních technologií fungujících v bankovním sektoru. Protože hodnocení jakosti software není zcela běžnou praxí, nebude bez zajímavosti odhadnout, pro koho bude navržená metoda určena, resp. pro které pracovní pozice může být relevantní. Audit informačního zabezpečení a hodnocení jakosti software může být prováděno z různých úhlů pohledu zájmových skupin. Každá tato skupina má jiná kritéria hodnocení jakosti softwarových produktů. Tato skutečnost je též motivace pro vytvoření dynamické metody stanovení množiny prvků měření kvality v konkrétní situaci. Navržená metoda tento předpoklad různosti pohledů na základě stanovené cílové skupiny následuje, konkrétně v podobě výběru expertů pro sběr požadavků na jakost.

Autor této práce se tématem hodnocení jakosti informačních systémů teoreticky zabývá od bakalářského stupně studia vysoké školy na Katedře informačního inženýrství Provozně ekonomické fakulty České zemědělské univerzity v Praze, pod vedením prof. Vaníčka. Tato problematika je, zejména díky prof. J. Vaníčkoví, na katedře vyučována a zkoumána, a zároveň má své další pokračovatele.

1.1 Cíle disertační práce

Původní cíl disertační práce zabývat se obecným modelem vlivu jakosti informačního zabezpečení na hospodářský výsledek podniku byl na základě doporučení nového školitele, doc. Tomana, zúžen a jako klíčový dílčí cíl bylo identifikováno vytvoření metody a měr pro kvantifikovaný přístup k řízení jakosti. Na základě tohoto doporučení, a po schválení oborovou radou, byl upraven i název disertační práce, aby lépe vystihoval řešenou problematiku.

Hlavním cílem disertační práce je tedy vytvoření metody a měr pro kvantifikovaný přístup k řízení jakosti softwarových produktů. Metoda je způsob, jakým rigorózně stanovit potřebné atributy k měření, tj. atributy v modelu. Výsledkem metody je model, který je určen pro použití k hodnocení jakosti software a jako podpora auditu informačních systémů.

Od hlavního cíle práce jsou odvozeny dílčí cíle práce, a to:

- Aplikovat navrženou metodu v reálném prostředí a ověřit tak její použitelnost a vytvořit výsledný model prvků měření kvality, který je platný pro konkrétní implementovaný informační systém.
- Navrhnout některé prvky měření kvality. Podle požadavků na jakost, resp. podle expertních odhadů, navrhnout a popsat jednotlivé prvky měření kvality.
- Analýza dostupných pramenů projektu SQuaRE a kriticky zhodnotit tento projekt. Práce obsahuje analýzu SQuaRE a překlad klíčových konceptů, které vychází z originálních textů norem a český překlad není zatím vytvořen.

1.2 Metodický postup

Při zpracování disertační práce byla použita kombinace analýzy teoretických vstupů a empirické metody výzkumu. Analýza teoretických vstupů práce je použita pro formulaci navrženého řešení. Empirické metoda následně pomohla ověřit použitelnost vytvořeného řešení v reálném prostředí.

První část práce má rešeršní charakter. Pro její zpracování, a současně vznik a realizaci této disertační práce, je detailně se seznámit s problematikou jakosti softwarových produktů, zejména informačních systémů, a jejím hodnocením. Dalším teoretickým vstupem je popis současného stavu mezinárodní normalizace jakosti software a dle dostupných pramenů jeho kritické zhodnocení. Analýza je zaměřena zejména na probíhající projekt SQuaRE. Jako nejdůležitější z této části autor považuje analýzu konceptu prvků měření kvality, kterému je věnována jedna celá kapitola. Hodnocení jakosti softwarových produktů je významnou částí teoretického přehledu práce. Tuto část doplňuje charakteristika způsobu měření jakosti. Rovněž kapitola audit informačních systémů a způsob jeho provádění autor považuje za důležitou součást rešerše.

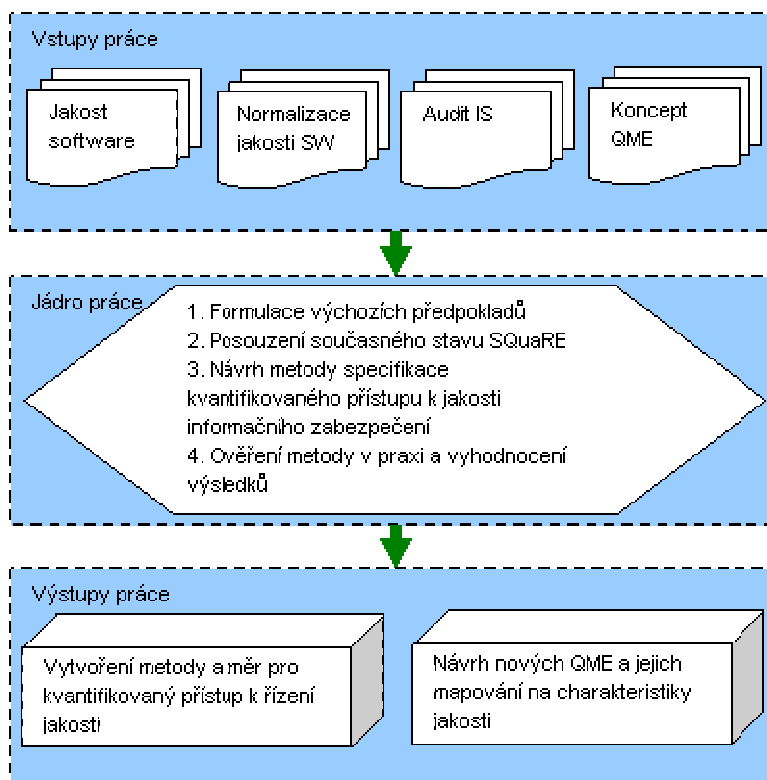
Autor si je vědom relativně velkého rozsahu teoretické, rešeršní části disertační práce. Uvedené tematické celky však považuje za důležité z hlediska komplexnosti práce a předkládaného řešení cílů uvedených v kapitole 1.1. V některých případech je záměrně ponechán originální text a popisky bez překladu do českého jazyka.

Po prostudování dostupných pramenů, které zahrnují jak knižní tituly, příspěvky konferencí, online zdroje a mezinárodní standardy, byly autorem identifikovány a formulovány výchozí předpoklady nastíněné v Úvodu práce, ze kterých autor vycházel

při návrhu vlastní metody. Autor rovněž zužitkoval své praktické zkušenosti jak z liniového, tak procesního a projektového řízení v prostředí banky.

Stěžejní část práce obsahuje jádro řešení, které je formalizováno do podoby vlastního návrhu metody. Na základě teoretických poznatků uvedených v rešeršní části práce a zkušeností autora z praxe bude tedy popsán návrh metody pro kvantifikovaný přístup k hodnocení jakosti IS. Navržená metoda byla aplikována v praxi a jejím výsledkem je model, který je optimalizací expertních odhadů a jejich požadavků na jakost software. Způsob vytvoření výsledného modelu je také v této části práce popsán.

Závěrečná část práce má empirickou povahu. Ověření výsledků autor provedl ve finančním sektoru. Pomocí experimentu na zvoleném typu implementovaného softwarového produktu a současně výběrem vhodných expertů pro sběr dat autor ověřuje aplikovatelnost navržené metody v konkrétním prostředí. Vyhodnocení výsledků výzkumu pak potvrdí, nebo vyvrátí (částečně nebo úplně) formulované předpoklady. Na základě výsledků lze předpokládat, že navržená metoda bude použitelná i pro další typy podniků a odvětví. Samotná struktura disertační práce je uvedena na obrázku 1.



Obrázek 1: Struktura disertační práce. Zdroj autor.

1.3 Použitá terminologie

Pro účely tohoto dokumentu jsou použity termíny a definice uvedené v ISO/IEC 25000, ISO/IEC 25010, ISO/IEC 25020, ISO/IEC 15939 a International Vocabulary of Basic and General Terms in Metrology. Další použité termíny a zkratky jsou vysvětleny individuálně v textu.

1.4 Struktura disertační práce

Struktura práce je koncipována do 8 kapitol:

První kapitola je věnována úvodu, představení a zdůvodnění vzniku této práce. Dále jsou popsány hlavní a dílčí cíle disertační práce a metodický postup řešení – dosažení výsledků. Je zde zmíněn také odkaz na použitou terminologii. Vlastní struktura práce je věnována jedna podkapitola, která rovněž obsahuje grafickou podobu struktury. Tématem dalších podkapitol je stručná charakteristika současného stavu řešení dané problematiky a přehled souvisejících tematických celků, které nejsou obsahem této disertační práce.

V druhé kapitole je popsána teorie jakosti informačního systému. Text je vztažen k současnému chápání a posuzování jakosti softwarových produktů, zejména pak k různým pohledům na jakost software.

Ve třetí kapitole práce je popsán současný stav mezinárodní normalizace jakosti softwarových produktů a další vývoj v této oblasti. Charakterizovány jsou platné normy, však hlavním tématem kapitoly je představení projektu – připravované koncepce mezinárodních norem pro jakost software – modelu SQuaRE. Součástí kapitoly jsou ještě charakteristiky a atributy jakosti software.

Čtvrtá kapitola je věnována teorii měření jakosti informačního systému. Charakteristice postupu měření jakosti předchází podkapitoly, kde jsou uvedeny vlastnosti a požadavky na software. Tuto kapitolu uzavírá popis základních užívaných měřicích stupnic.

V páté kapitole práce je uveden současný stav provádění auditu informačního systému, důraz je kladen zejména na obecný postup jeho provádění.

V šesté kapitole je detailně popsán princip tzv. prvků měření kvality (QME). Po představení tohoto konceptu je uvedena základní sada těchto prvků vycházející

z projektu SQuaRE. Pak je podrobně charakterizován postup (tvorba) projektování prvků měření kvality

Sedmá, pravděpodobně stěžejní kapitola práce, je věnována vlastnímu návrhu metody a měr pro kvantifikovaný přístup k řízení jakosti. Definovány jsou výchozí předpoklady pro vznik metody. Po stanovení těchto předpokladů je navržena vlastní metoda pro kvantifikovaný přístup k jakosti softwarových produktů. Tato metoda je uplatněna v praxi a na základě získaných výsledků a jejich vyhodnocení jsou potvrzeny uvedené předpoklady. Výsledkem metody popsaným v kapitole je model pro podporu hodnocení jakosti IS.

Osmá kapitola obsahuje shrnutí disertační práce a interpretaci dosažených výsledků. Dále jsou navrženy další možnosti výzkumu a využití výsledků práce.

Na konci disertační práce je uveden přehled použitých pramenů v této práci, seznam zkratk, seznam obrázků, seznam tabulek a jednotlivé přílohy, které nebyly zařazeny přímo do struktury (kapitol) práce.

1.5 Omezení a limity práce

Autor disertační práce předpokládá, že navrhovaná metoda, která bude dále detailně popsána, najde své uplatnění přes zamýšlenou univerzálnost spíše v menších firmách. Její aplikace při korektním použití nepředstavuje vysoké nároky na zdroje (finance, lidské zdroje) a její provádění pro svoji relativní jednoduchost negeneruje další významné náklady pro danou firmu. Metoda nemá ambice být konkurencí pro profesionální nástroje řízení podnikové informatiky (jako je např. [COBIT 2007], apod.) používané ve velkých společnostech, ale být spíše jejich doplňkem specializovaným na jakost IS.

Tato disertační práce se omezuje na návrh a ověření metody pro kvantifikovaný přístup k řízení jakosti. Přestože je záměrem, aby tato metoda podporovala hodnocení jakosti a případně audity informačních systémů, a vlastní proces hodnocení jakosti a auditu IS je teoreticky v rešeršní části popsán, provedení těchto aktivit není obsahem práce.

Autor návrhu metody předpokládá, že dále popsaná metoda není určena pro každého (běžného) uživatele informačních technologií. Pravděpodobně bude třeba, aby

uživatel metody měl zkušenosti rovněž s procesním, nebo projektovým řízením. V ideálním případě znalý problematiky hodnocení jakosti, případně postupů auditu IS.

1.6 Současný stav řešení dané problematiky

Hodnocení je systematické posouzení hodnoty nebo významu nějakého objektu. Jinými slovy, evaluace je systematické získávání a posuzování informací, které poskytují užitečnou zpětnou vazbu o nějakém objektu. Obecný cíl většiny evaluací je poskytnout "užitečnou zpětnou vazbu" pro různé strany, např. sponzory, klientské skupiny, zaměstnance, apod. Nejčastěji je zpětná vazba vnímána jako užitečná, pokud pomáhá při rozhodování. Evaluace lze rozlišovat podle toho, zda má sloužit vědeckým účelům (týká se např. účinnosti metod a prostředků), nebo praktickým účelům (týká se konkrétního procesu, instituce, apod.). Při evaluaci se využívá široká škála kvantitativních i kvalitativních metod a technik, např.: dotazníkové šetření, experiment, kvalitativní/kvantitativní analýza, statistika, případová studie atd. [EVAL 2006].

Moderní interní audit se začal vytvářet v roce 1941, kdy vznikl Institut interních auditorů. Tehdy rozšířili interní auditoři svou působnost na hodnocení všech operací probíhajících v organizaci. Od tohoto období se tak postavili na úroveň svých externích protějšků [Svatá 2011].

V prvních vývojových etapách se interní audit zabýval převážně finančním auditem a vnitřní účetní kontrolou, postupně se však přidávaly další funkce. Vzhledem k zaměření interního auditu na přidávání hodnoty a dosažení vysoké efektivnosti veškerých činností se klade velký důraz na audit výkonu [Svatá 2011].

Podle definice Mezinárodního institutu interních auditorů je interní audit nezávislá, objektivní, ujišťovací a konzultační činnost zaměřená na přidávání hodnoty a zdokonalování procesů v organizaci. Ujišťovací činnost v tomto případě znamená objektivní prověření a zhodnocení průkazného materiálu s cílem poskytnout managementu organizace nezávislé hodnocení systému řízení rizik, zavedených řídicích a kontrolních systémů a mechanismů, nebo stylu řízení a správy organizace. Podle stanovených cílů interního auditu, kterých má být auditem dosaženo, jsou prověřovány různé činnosti organizace. Tomu je uzpůsoben i typ interního auditu [Svatá 2011].

Obsah auditu IS, respektive záběr auditu prošel vývojem, který lze dle [Svatá 2011] stručně shrnout do těchto etap:

1. etapa hodnocení výsledků zpracování na počítačích – obsahem auditu jsou vstupy, zpracování a výstupy informačních systémů, důraz je na prověřování minulých a existujících transakcí,
2. etapa hodnocení rizika a kontrol počítačového zpracování – obsah se rozšiřuje na preventivní hodnocení spolehlivosti a průkaznosti počítačově zpracovávaných transakcí např. již v etapě jejich vývoje nebo pořizování,
3. etapa hodnocení a optimalizace procesů IT i navazujících procesů – obsah se dále rozšiřuje z oblasti čistě „počítačové“ do oblastí souvisejících (např. postavení útvaru IS/IT v rámci organizace, realizace přidané hodnoty k business procesům pomocí IS/IT, hodnocení metrik služeb, uzavírání smluv apod.).

Pro oblast hodnocení jakosti informačních systémů existují dostupné prameny zejména v podobě mezinárodních norem a technických zpráv. Tyto normy vytváří mezinárodní normalizační organizace ISO spolu s mezinárodní elektromechanickou komisí IEC v gesci společného výboru ISO/IEC JVC 1 Informační technika, speciálně jeho podvýboru SC7 – Systémové a softwarové inženýrství, přičemž hodnocení kvality se věnuje především jeho pracovní skupina WG 6. Normy připravené v těchto orgánech procházejí třemi úrovněmi hlasování členských zemí ISO/IEC JTC 1. Po publikaci sekretariátem ISO jsou automaticky přebírány evropskými normalizačními institucemi CEN a CENELEC jako evropské normy. Z přístupových dohod České republiky k Evropské unii plyne závazek převzít je i jako české normy. [Vaníček 2008c].

Lze konstatovat, že v podmínkách ČR se normy, o kterých tento příspěvek informuje, využívají jen velmi málo. To je dle Vaníčka [Vaníček 2004b] snad způsobeno tím, že u nás v oblasti inforatických produktů stále převládá trh dodavatele nad trhem zákazníka. Iniciativu ke změně nelze očekávat od dodavatelů. Ve vyspělých státech jsou dodavatelé pak nuceni k prohlášením o shodě s normou a k závazkům pro případ, kdy se ukáže, že toto prohlášení bylo chybné. Jsou trhem nuceni i k tomu, aby si shodu nechali na své náklady ověřit nezávislou atestační autoritou.

Mezinárodní normy a technické zprávy vydávané normalizačními autoritami bývají výsledkem kompromisu mezi vývojáři, dodavateli a uživateli software a zástupci vědecké komunity. Vlivem nutného několikaúrovňového schvalování těchto norem v příslušných normalizačních orgánech normy často již v okamžiku publikace neodrážejí aktuální stav poznatků v oboru informačních technologiích, který se velmi rychle rozvíjí. [Kardoš 2010].

1.7 Související témata, která nejsou obsahem práce

Problematika jakosti softwarových produktů je relativně velmi rozsáhlá, zasahuje do různých dalších oblastí, jako je např. řízení podnikové informatiky a její výkonnosti, atd. Z tohoto důvodu je třeba do obsahu této práce záměrně nezařadit některé tematické celky, které s touto částí informatiky souvisí, ale neovlivňují dosažení cílů disertační práce.

Jsou to především:

- detailní analýza nástrojů pro řízení podnikové informatiky a řízení podnikové informatiky; této problematice je věnována např. publikace [Novotný et al. 2011],
- detailní analýza řízení výkonnosti podniku, která je detailně rozpracována např. v monografiích [Král et al. 2006] a v [Synek et al. 2006].
- detailní popisy procesů životního cyklu software dle [Gála et al. 2009],
- typologie a architektura informačních systémů, které jsou popsány např. v monografiích [Gála et al. 2009], [Vrana, Richta 2005].

2 Jakost informačního systému

Před tím, než bude rozebírána jakost IS/IT, je snad na místě krátká definice informačního systému: Informační systém je dle [Tvrđíková 2000] soubor lidí, metod a technických prostředků zajišťujících sběr, přenos, uchování, zpracování a prezentaci dat s cílem tvorby a poskytování informací dle potřeb příjemců informací činných v systémech řízení.

Podle [Vaníček 2004a] citované základní normy ISO 9000 pro jakost jsou **jakost a kvalita synonyma¹** (tak jsou chápána také v této práci), **vymezená jako stupeň splnění požadavků souborem inherentních znaků**. Slovo „inherentní“ je zde třeba chápat jako trvalý, objektivně existující znak hodnoceného předmětu, tedy ne znak, který mu byl dodatečně přiřazen. **Jakost produktu je souhrn podstatných vlastností produktu, které určují míru uspokojení daných (obecně očekávaných) [implied] a stanovených [stated] potřeb [needs] uživatele produktu, v případě užití produktu stanoveným způsobem**. Tyto potřeby jsou vyjádřeny formou pokud možno exaktních požadavků na produkt.

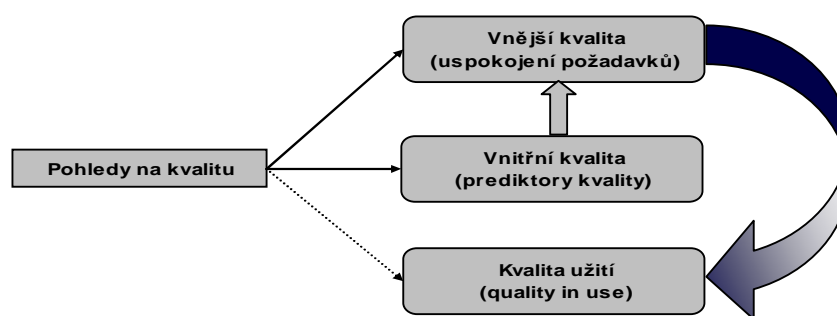
Kvalita jakéhokoliv produktu je definována jako souhrn jeho vnitřních vlastností směřujících k zabezpečení požadavků všech zúčastněných stran na produkt. Tyto požadavky by měly vycházet ze skutečných potřeb všech účastníků, především pak uživatelů produktu. Pro objektivní hodnocení jakosti je důležitá úloha normalizace a měření [Vaníček 2008].

Jakost můžeme posuzovat [Vaníček 2004a] buď na základě zkoumání produktu samotného nebo na základě používání tohoto produktu. Podle toho, jak k jakosti přistupujeme, hovoříme někdy o tak zvané „vnitřní jakosti“ nebo o „vnější jakosti“. V mezinárodních normách se kromě toho vyskytuje i pojem „jakost užití“, někdy překládaný i jako „jakost při používání“. Softwarový produkt nemůže být k uspokojení potřeb užit nikdy sám o sobě, ale vždy pouze v rámci nějakého systému, který zahrnuje i hardware, organizační opatření a lidi, které s ním pracují. Jde pouze o to, jakost čeho je předmětem našeho zkoumání. Striktně vzato jakost je vždy vlastností celého systému, protože pouze ten, jako celek, uspokojuje naše potřeby. Pouze působení systému jako celku lze zkoumat. Při tomto pojetí:

¹ Pojmy jakost i kvalita se v disertační práci vyskytují v různých modifikacích, která vychází z ustálených spojení a překladů standardů.

- **Vnitřní jakost** [*Internal quality*] je souhrn podstatných vlastností produktu, které určují jeho schopnost uspokojovat stanovené a dané potřeby při používání za stanovených podmínek.
- **Vnější jakost** [*External quality*] je rozsah uspokojování stanovených a daných potřeb příslušným produktem při jeho používání za stanovených podmínek.
- **Jakost užití** [*quality in use*] je souhrnem toho, jak systém jako celek uspokojuje potřeby uživatelů.

Pro podporu péče o kvalitu jsou známy a široce využívány normy řady ISO 9000 zaměřené na management kvality. Tyto normy jsou orientovány především na proces vývoje, výroby a dodávek produktů, kterými mohou být výrobky hmotné i nehmotné povahy, služby a kombinace těchto produktů. Představují obecné požadavky na tyto procesy, platné pro nejrůznější druhy produktů. Pro zákazníka mají ten význam, že pokud je vývojář, výrobce a dodavatel dodržují, je vysoká míra záruk, že produkt bude kvalitní a uživatele uspokojí. Samozřejmě není možné stanovit zcela obecná pravidla, v čem spočívá kvalita produktu tak, aby ji bylo možné hodnotit pouze z produktu samého, bez informací o tom, jak produkt vznikl. Taková pravidla se nutně liší případ od případu, podle konkrétního typu hodnoceného produktu. Nicméně pro zákazníka je důležité takováto pravidla mít, protože o procesu vzniku produktu často má jen velmi neúplné informace [Vaníček 2008a].



Obrázek 2: Model kvality software. Zdroj [Vaníček 2004a].

Při rozhodování o nákupu produktu nebo výběru z nabídky není konečná volba provedena pouze na základě hodnocení jakosti. Při volbě mezi alternativami hraje jakost sice důležitou roli, nikoliv však jedinou. Obvykle objem disponibilních prostředků či požadavek rentability pevně limituje shora cenu, kterou můžeme za produkt zaplatit. Pokud však tento limit dosažen není, je naše rozhodování mezi alternativami problémem optimalizace poměru *jakost / cena*. Přitom obvykle tyto dva faktory působí protichůdně. Čím vyšší jakost, tím vyšší i cena. Právě tak, jako je volba omezena

cenovým limitem shora, měl by být vždy stanoven limit zdola na nejnižší možnou, ale ještě přijatelnou, jakost. Pokud tento limit dodržen nebude, je to signál, že naše potřeby nebudou uspokojeny ani v úrovni, ze které již není rozumné „slevit“. V tom případě bychom měli uvažovat o jiném řešení. Problém při optimalizaci poměru *jakost/cena* spočívá především v tom, že cenu lze hodnotit snadno v poměrové stupnici (víme co je to poloviční nebo o daný počet procent nižší či vyšší cena), hodnocení jakosti však bývá obtížné (např. vyšší jakost o 10%), obvykle i nemožné. Kromě jakosti a ceny mohou brát pro výběr alternativ roli i jiné faktory, například strategická orientace budoucího uživatele a jeho zájem na spolupráci s určitým okruhem dodavatelů [Vaníček 2004a].

Při prodeji softwarových produktů a služeb na trhu se setkávají dva do jisté míry protichůdné zájmy:

- zájmy výrobce a prodejce maximalizovat svůj zisk,
- zájem kupujícího uspokojit své potřeby na co nejvyšší úrovni (s maximální jakostí) při minimálním vynaložení prostředků.

Protichůdnost je však v dobře fungující tržní společnosti do jisté míry jen zdánlivá. Snaha zabezpečit pouze prvý cíl, bez ohledu na druhý, může mít úspěch jen krátkodobě. Pokud prostředí není monopolní, je nutné si zákazníka získat. To znamená splnit jeho požadavky na jakost a nabídnout produkt za přijatelnou cenu. Jakost se tak stává hybnou silou pokroku. Péče o jakost by měla být součástí globální strategie každého ekonomického subjektu, který chce být dlouhodobě úspěšný. Pro úspěch v konkurenci je třeba znát a respektovat především [Vaníček 2004a]:

- Potřeby potenciálních zákazníků, včetně odhadu jejich požadavků na jakost.
- Nejen zájmy „přímých zákazníků“, ale i zájmy jejich okolí, všech zainteresovaných stran, tedy v podstatě celé společnosti. Jen tak je totiž možné vybudovat a udržet „dobré jméno“ firmy.
- Nabízet produkty za ceny přijatelné pro zákazníka a přitom získat a zachovat přiměřený zisk.

2.1 Jakost produktu a jakost procesu

Jakost je třeba hodnotit u **procesu** i u **produktu**. **Proces** je (ISO 9000) definován jako **soubor vzájemně souvisejících a vzájemně působících činností, který přeměňuje vstupy na výstupy**. **Produkt** je **výsledek procesu**.

V současné době dle [Vaníček 2004a] existují dva paralelní přístupy k problematice hodnocení jakosti.

- Prvý vychází z myšlenky stanovit požadavky na řízení procesu tvorby produktu, které povedou k tomu, že jeho výsledek – produkt bude mít požadovanou úroveň jakosti. Tyto postupy jsou poměrně dobře propracované a známé. Reflektují je již několikrát zmiňovaná norma „ČSN EN ISO 9000 ed. 2“ z roku 2002, „Systémy managementu jakosti – Základy, zásady a slovník“ a řada norem, které na ni navazují.
- Druhý vychází z myšlenky hodnotit jakost z hlediska uživatele či opatřovatele produktu, kterého zajímají pouze vlastnosti daného produktu a do jaké míry uspokojí jeho potřeby.

Souvislost mezi normami na řízení procesů z hlediska jakosti a normami pro jakost produktu je zřejmá. Vychází se z obtížně zpochybnitelného paradigmatu, že pokud je při vývoji a výrobě ve firmě pořádek, bude výrobek jakostní. Pokud je vývoj a výroba živelná a bez pravidel, nelze jakost výstupů nijak garantovat.

Hlavní rozdíl mezi těmito normami a normami na jakost výrobku spočívá v tom, že normy pro jakost procesu sledují především zájmy vývojářů a výrobců, zatímco normy pro jakost produktu sledují spíše zájmy odběratelů a uživatelů. V současné době jsou normy pro jakost procesu v našem prostředí známější a používanější než normy pro jakost produktu. V našich podmínkách v oblasti informačních a komunikačních technologií stále převládá „trh dodavatele“ před „trhem zákazníka“. Vývojáři a výrobci mají především zájem zlepšit svou organizaci a zvýšit zisk. Přímé doporučení jak měnit proces je tedy pro ně cennější než nedobrá „známka“ za výrobek, bez přímé vazby na konkrétní nápravu.

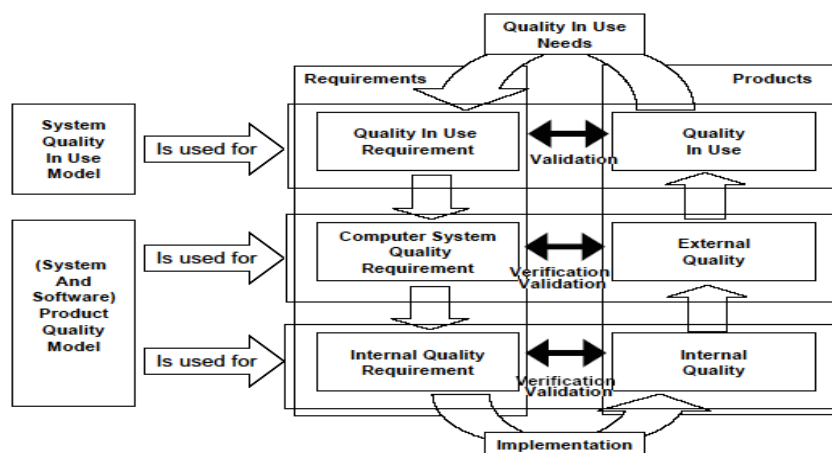
2.2 Jakost etapy životního cyklu

Poslání péče o jakost je dle [Vaníček 2004a] získat s relativně nízkými náklady produkt, který by měl vysokou úroveň jakosti. Jakost je míra, v které produkt uspokojuje potřeby uživatelů. Lze ji proto stanovit až po ukončení vývoje. To však bývá již pozdě na nápravu. Jakákoliv změna v této fázi je již velice drahá. Je proto žádoucí odhadovat budoucí jakost co nejdříve. Protože před dokončením vývoje nelze měřit atributy hotového produktu, je třeba měřit a hodnotit atributy výsledků jednotlivých etap vývoje,

u kterých je předpoklad, že budoucí jakost ovlivní. Tyto atributy jsou nazývány prediktory.

K tomu, aby bylo možné i v těchto počátečních etapách postupovat exaktně, je nezbytné přesně, pokud možno formálně, vymezit, jaké mají být výsledky těchto etap životního cyklu. Při vývoji softwaru se vyplatí dodržovat následující zásadu pro jeho zkoušení (testování): „Testuj ihned, jakmile je co testovat“. Pro měření a hodnocení jakosti lze tento slogan rozšířit na „Jakost je třeba hodnotit, jakmile je co hodnotit“.

Z modelu kvality životního cyklu software rovněž vyplývá, že dosahování přijatelné úrovně kvality by mělo být nedílnou součástí procesu rozvoje pro každý typ kvality, včetně: požadavky, implementace a ověřování výsledků. Na obrázku 3 je znázorněn model kvality a životního cyklu software [ISO/IEC 2009a]:



Obrázek 3: Kvalita a životní cyklus software. Zdroj [ISO/IEC 2009a].

2.3 Různé pohledy na jakost

Typickým případem produktů, kde seriózní hodnocení jakosti je mimořádně nesnadné, jsou právě produkty z oblasti informačních technologií. Hodnocení jakosti vyžaduje personál s vysokou mírou znalostí a zkušeností. Často vyžaduje i poměrně nákladné technické zázemí. Ne vždy je pro budoucího uživatele rozumné, aby si hodnocení prováděl sám. Někdy je mnohem efektivnější hodnocení svěřit nezávislým subjektům, které je za úplatu provádějí profesionálně. Tyto subjekty mohou být mnohem lépe materiálově i personálně vybaveny. Vývoj ve světě jednoznačně směřuje k tomu, svěřovat hodnocení jakosti takovýmto testovacím střediskům. Ve vyspělých ekonomikách tato střediska dokonce pracují převážně na zakázku firem, které produkty nabízejí. Méně často na zakázku potenciálních kupců a uživatelů. Bez získání

odpovídajícího atestu jakosti od testovacího střediska (zpravidla akreditovaného pro tuto činnost nějakou autoritou – například státní) se totiž podstatně snižují šance produktu na trhu. Lze tedy vymezit následující přístupy a pohledy na hodnocení jakosti [Vaníček 2004a]:

- přístup obstaravatele,
- přístup projektanta,
- přístup nezávislého hodnotitele.

Obstaravatelem nebo též akvizitérem [*acquirer*] je míněn zákazník, který je buď budoucím uživatelem produktu nebo produkt poptává s tím, že jej použije ve svém produktu jako „polotovár“. Jde především o to, kdo potřebuje jakost vyhodnotit, aby provedl výběr alternativ, nebo rozhodl o koupi produktu. Typickým příkladem akvizitéra v oblasti informačních technologií je systémový integrátor. Obstaravatel může hodnotit jakost produktu až po dokončení jeho vývoje. Nemá zpravidla přístup k veškeré vývojové dokumentaci. Měl by však mít zabezpečenu možnost přístupu ke všem informacím, které jsou významné pro jeho rozhodování, a to zdarma a dříve, než se o akvizici definitivně rozhodl. Tato práva by mu měly technické normy garantovat. Jak daleko mají tato práva jít, je ovšem otázka kompromisu mezi ochranou oprávněných zájmů akvizitéra a ochranou oprávněných zájmů prodávajícího, jako je například obchodní tajemství či utajení konkrétního technického řešení či výrobního postupu.

Projektant [*developer*] je ten, kdo systém vyvíjí. Má na jakosti také zájem, může o ni však pečovat již dříve. I on samozřejmě musí vycházet z potřeb budoucích zákazníků a zainteresovaných stran. Na začátku vývoje by měl jako součást specifikace vymezit i požadavky na jakost. Tu by pak měl prověřovat ve všech etapách vývoje. V počátečních etapách, pokud produkt nebude ještě funkční, nebude mít samozřejmě možnost jakost testovat přímo. I tehdy však bude moci prověřovat určité atributy „polotovarů“, či průběžných etap řešení, u kterých je předpoklad, že mohou sloužit jako prediktor výsledné jakosti produktu. To, které atributy je vhodné hodnotit a jaký je jejich vliv na výslednou jakost, je samozřejmě podstatná otázka. Na rozdíl od akvizitéra lze předpokládat, že projektant má přístup ke všem informacím, které mohou být pro hodnocení jakosti relevantní.

Nezávislým hodnotitelem [*evaluator*] může být právnická či fyzická osoba, která není přímo svázána ani s vývojem ani akvizicí hodnoceného projektu. Může jím však být i relativně nezávislý útvar uvnitř organizace firmy, která provádí vývoj.

Někteří „velcí“ odběratelé mají taková testovací střediska uvnitř své organizační struktury. Důležité je, že zájem takového hodnotitele musí být zřetelně oddělen od zájmů projektanta i akvizitéra. Důvodem vzniku takovýchto nezávislých testovacích středisek je soustředění specialistů a potřebné techniky, a tím i jejich efektivnější využití. Nezávislý hodnotitel má samozřejmě přístup ke všem informacím, ke kterým má přístup akvizitér. Často má však přístup i k informacím dalším, které mu hodnocení poskytne. Skutečnost, že prohlášení o tom, na základě jakých vstupů bylo hodnocení provedeno, bývá součástí hodnotícího výroku, totiž projektanty nutí k tomu, aby tyto informace nezávislému hodnotiteli poskytli. Neposkytnutí těchto informací může být totiž posuzováno negativně. Na druhé straně projektant snáze získá od nezávislého hodnotitele záruky za to, že poskytnuté informace nebudou zneužity.

V poslední době roste i zájem projektantů o to, aby nezávislí hodnotitelé hodnotili nejen výsledný produkt, ale přímo proces, ve kterém je produkt vyvíjen, případně vyráběn. Je totiž předpoklad, že pokud je tento proces „v pořádku“, nemůže vzniknout nejakostní produkt. Takováto inspekce může být pro projektanty dokonce užitečnější než pouhé „oznámkování“ hotového výrobku, při kterém „špatná známka“ je pouze důvodem k roztrpčení, nezahrnuje však cestu k nápravě. Takovéto hodnocení procesu přináší přímo již i podněty k nápravným opatřením, které směřují k zvýšení jakosti. Rozhodovat má vždy ten, kdo za důsledky rozhodnutí ponese odpovědnost. Snaha mechanicky přenést odpovědnost na jakýkoliv „poradenský subjekt“, který neponese následky realizace svého doporučení, by tedy bylo velmi krátkozraké. Důležité je samozřejmě, aby hodnotitel byl skutečně nezávislý a nehrozil žádný střet zájmů. Nesmí pochopitelně sám nabízet produkt podobného typu. Důležité je mít i záruky za to, že hodnocení bylo provedeno kvalifikovaně [Vaníček 2004a].

3 Mezinárodní normalizace jakosti software

Slovem **norma** [*standard*] je označena technická norma. Ta je vymezena jako **společně dohodnutý předpis pro technický nebo technicko ekonomický stav nějaké entity nebo průběh nějakého jevu za daných podmínek** [Vaníček 2004a]. Normy umožňují sjednotit pohled na danou problematiku a vytvořit jednotně chápaná pravidla. Usnadňují i kombinovat produkty různých dodavatelů a tím podporuje konkurenční prostředí. Zároveň je patrné, že s rostoucí globalizací lidské společnosti roste důraz na to, aby normalizace byla co nejširší. Důležité jsou proto především ty normy, které jsou přijaty celosvětově. V oblasti informačních systémů a informačních technologií se naštěstí celosvětové normy přebírají v evropské i české normalizační činnosti bez jakýchkoliv změn. Nebezpečnost norem spočívá především v tom, že norma ze své podstaty fixuje jakýsi pevný stav. Informatika je však oborem, který se bouřlivě vyvíjí. Dogmatické lpění na normě se tak může stát i brzdou pokroku. Zahnutí nových vědeckých poznatků do norem znesnadňuje i požadavek, aby norma byla srozumitelná co nejširšímu okruhu uživatelů. To u nových poznatků vědy bývá často obtížné zajistit. Přijetí normy vyžaduje široký konsensus zúčastněných zemí a příprava textu normy a její schvalování v několika krocích je časově náročné. Nezřídka se tak stává, že norma bývá přijata příliš pozdě. Dodatečné přizpůsobení se normě bývá náročné a v případě již širokého rozšíření produktů, které jí nevyhovují, i nereálné.

U mezinárodních norem [*International standard – IS*] z oblasti informačních technologií se ustálila následující stavba kapitol [Vaníček 2004a]:

- **Předmluva** [*Foreword*] není číslována a obsahuje nenormativní informace o tom, jak a proč dokument vznikl, jak byl schvalován, případně další důležité historické souvislosti.
- **Úvod** [*Introduction*] není číslován, nemá normativní charakter a obsahuje výklad problematiky, kterou se norma zabývá, případné důvody pro volbu toho řešení, které je v normě uplatněno.
- **Předmět normy** [*Scope*] má číslo 1. Je to důležitá část normy, která přesně vymezuje, na co se norma vztahuje a komu je určena. Mimo vymezený předmět normy není legitimní požadovat její aplikaci.
- **Shoda** [*Conformance*] mívá číslo 2. Obsahuje tak zvaný **výrok o shodě** [*conformance clause*], který určuje, respektive shrnuje, co musí být především

splněno, aby bylo možné konstatovat, že vyšetřovaná entita je ve shodě s normou.

- **Normativní odkazy** [*Normative references*] mívá číslo 3 a uvádí odkazy na jiné normy, které na příslušnou normu navazují, její ustanovení doplňují nebo zavádějí pojmy užívané v dané normě.
- **Definice** [*Definitions*] mívá číslo 4. Tato kapitola je závazná. Vymezuje užití pojmy. Nejde zpravidla o přesné definice, ale o neformální výklad pojmů. Pokud pro danou oblast existuje terminologická norma, uvádějí se zde pouze pojmy, které v terminologické normě definovány nejsou, případně ty, kde je třeba pojem chápat odlišně než v jiných normách.
- **Další kapitoly** jsou členěny a pojmenovávány podle vyšetřované problematiky. Počet dalších kapitol závisí výhradně na struktuře problémové oblasti a na dodržení požadavku, aby dokument byl přehledný. Jsou obvykle číslovány počínaje číslem 5.
- **Přílohy** [*suplement*] - bývají číslovány velkými písmeny anglické abecedy (A, B, C, ...). Přílohy mohou být **závazné** [*normative*], pokud obsahují nějaké závazné požadavky nebo zavádějí nové pojmy, a nebo **informativní** [*informative*], pokud slouží pouze k bližšímu vysvětlení situace.

3.1 Současný stav

V oblasti související s informační a komunikační technikou působí na celosvětové úrovni dvě mezinárodní organizace pro normalizaci [Vaníček 2004a]:

- **ISO** (*International Organization for Standardisation*), která se zabývá normalizací obecně a
- **IEC** (*International Electromechanical Commission*), která se zabývá normalizací v oblasti elektrotechniky.

Oblast informačních technologií je na pomezí působnosti těchto dvou organizací. Aby práce nebyly zbytečně dublovány, zřídily obě tyto organizace společný technický výbor s názvem **ISO/IEC JTC1 Informační technologie** [*Information Technology*]. Normy, které tato komise vytváří, se považují za normy ISO i IEC. Společný technický výbor JTC1 se dále dělí na řadu podvýborů [ISO/IEC 2009b]. Systémovými a softwarovými aspekty informačních systémů, včetně jakosti, se zabývá především podvýbor [ISO/IEC

2010] **ISO/IEC JTC1/SC7 Systémové a softwarové inženýrství** [System and Software Engineering]. Zde jsou otázky jakosti řešeny především v pracovní skupině:

- **WG6 Hodnocení a metriky** [Evaluation and Metrics] a
- **WG13 Struktura měření softwaru** [Software measurement framework].

Česká normalizace přebírá mezinárodní normy, které jsou v oblasti našeho zájmu totožné s Evropskými. Organizačně zajišťuje tvorbu a vydávání českých norem Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, který je ústředním orgánem státní správy, zastupovaným ve vládě ministrem průmyslu a obchodu. O normalizaci v oblasti státní a veřejné správy pečují příslušné útvary ministerstva vnitra. Na národní úrovni se odborně o normalizaci starají tak zvané technické normalizační komise, působící při ČSNI se statutem poradních orgánů. Nemají žádnou rozhodovací pravomoc a jejich vliv na vynakládání prostředků na normalizaci je mizivý. V oblasti IS/IT působí především komise:

- **TNK 20 Informační technologie a**
- **TNK 42 Výměna dat.**

Normy se připravují v pracovních skupinách expertů. Ti připravují a projednávají text norem, tak zvaný **pracovní návrh** [working draft – WD] neformálně. Každý člen pracovní skupiny hovoří sám za sebe, nikoliv za stát, který zastupuje. Poté, co pracovní skupina návrh uvolní, získá statut **návrhu výboru** [committee draft – CD]. Hlasují o něm země, které jsou plnoprávními členy příslušného podvýboru (SC). Pokud zde získá návrh kvalifikovanou většinu, stane se **návrhem mezinárodní normy** [draft international standard – DIS] a o něm probíhá další hlasování, tentokrát na úrovni národního zastoupení v JTC1. Je-li i zde schválen kvalifikovanou většinou, je vyhlášen jako norma. V obou kolech hlasování lze při kladném i záporném hlasu uplatnit připomínky. Při hlasování na úrovni DIS pouze formální, ne již věcné. Během této procedury se často hlasování opakují. Někdy bývá procedura ještě složitější a doplněná o další etapy tak zvaných konečných [final] verzí návrhů, FCD a FDIS. Norma se tak připravuje dosti dlouhou dobu. Po pěti letech její platnosti se přezkoumává, zda má být platnost ukončena, prodloužena nebo zda má dojít k novelizaci normy.

3.1.1 ISO/IEC 12119

První norma [Vaníček 2004a], kterou pracovní skupina WG6 vytvořila a ISO/IEC JTC1/SC7 schválila, má název **ISO/IEC 12119 Informační technologie – Softwarové**

balíky – Požadavky na jakost a zkoušení [*Information technology – Software packages – Quality requirements and testing*]. Byla přijata i jako česká norma se zkratkou ČSN před svým názvem. Týká se softwarových produktů, které jsou nabízeny a dodávány masově, širokému okruhu zájemců. Tato norma vymezuje pojem jakosti, přímo však neukládá dodavateli žádné povinnosti na dodržení určité její úrovně. Zavazuje jej pouze k tomu, aby poskytl potenciálnímu zákazníkovi poctivě všechny údaje předem. Zakládá povinnost vypracovat ke každému takto dodávanému balíku dokument s pracovním názvem „**Popis produktu**“. Důležitý je jeho obsah a to, že jej musí mít uživatel k dispozici zdarma nebo pouze za režijní náklady dříve, než se rozhodne, zda softwarový balík zakoupí či nikoliv. Tento dokument musí obsahovat vše, co je pro takovéto rozhodnutí podstatné. Dále musí obsahovat všechny nároky na hardware, software a na prostředí, nutné pro jeho funkci. Nesmí se stát, že po prostudování tohoto dokumentu uživatel zjistí až dodatečně, poté co zaplatil, že pokud chce zakoupený produkt efektivně používat, znamená to další nečekané výdaje na hardware nebo na pořízení novější verze operačního systému. Důležité je, že **veškerá tvrzení** tohoto dokumentu musí být **jednoznačná, navzájem nerozporná a testovatelná**. Obecná prohlášení reklamního typu se nepřipouštějí. Norma obsahuje minimální povinnou „osnovu“ tohoto dokumentu a pokyny pro to, jak lze splnění toho, co bylo v dokumentu deklarováno, ověřit zkouškou.

3.1.2 ISO/IEC 9126

Původní norma *ISO/IEC 9126 Informační technologie – Hodnocení softwarového produktu – Charakteristiky jakosti a jejich používání* byla již zrušena a nahrazena novou řadou ISO/IEC 9126. Ta obsahuje jednu normu a tři návazné technické zprávy [Vaníček 2004a]:

- Norma **ISO/IEC 9126-1 Informační technologie – Jakost softwarového produktu – Model jakosti** [*Information technology - Software product quality - Part 1: Quality model*]. Tato norma popisuje model jakosti. Byla převzata i jako česká norma ČSN ISO/IEC 9126-1.

Na ní navazují tři technické zprávy:

- **TR ISO/IEC 9126-2 Informační technologie – Jakost softwarového produktu – Část 2: Vnější metriky** [*Information technology - Software product quality - Part 2: External metrics*],

- **TR ISO/IEC 9126-3 Informační technologie – Jakost softwarového produktu – Část 3: Vnitřní metriky** [*Information technology - Software product quality - Part 3: Internal metrics*],
- **TR ISO/IEC 9126-4 Informační technologie – Jakost softwarového produktu – Část 4: Metriky² pro jakost užití** [*Information technology - Software product quality – Part 4: Quality in use metrics*].

Tyto zprávy obsahují návrhy jednotlivých typů měr pro atributy jakosti.

3.1.3 ISO/IEC 14598

Tato řada šesti norem je věnována postupům při hodnocení jakosti produktu. Všechny tyto normy byly převzaty i jako české normy překladem (se zkratkou ČSN před svým názvem). Všech 6 částí má společný název [Vaníček 2004a]:

ISO/IEC 14598 Informační technologie – Hodnocení produktů [*Information Technology - Software product evaluation*].

Jednotlivé normy pak za tento název doplňují:

- **Část 1: Obecný přehled** [*Part 1: General overview*],
- **Část 2: Plánování a řízení** [*Part 2: Planning and management*],
- **Část 3: Postup pro projektanty** [*Part 3: Process for developers*],
- **Část 4: Postup pro akvizitéry** (opatrovatele) [*Part 4: Process for acquirers*],
- **Část 5: Postup pro nezávislé hodnotitele** [*Part 5: Process for evaluators*],
- **Část 6: Dokumentace vyhodnocovacích postupů** [*Part 6: Documentation of evaluation modules*].

Prvá z uvedených norem zavádí terminologii a popisuje společné zásady. Druhá je věnována problému jak hodnocení jakosti organizovat. Další tři stanovují postupy při hodnocení ze tří hlavních pohledů: řešitelského, zákaznického a pohledu nezávislé třetí osoby, která se hodnocením zabývá. Poslední norma patří svým obsahem spíše do řady 9126. Pokouší se sjednotit dokumentaci postupů užívaných pro hodnocení. Určuje tedy pravidla jak jednotně popisovat míry, metody měření a jak jednotně zaznamenávat výsledky měření.

² V nových normách na měření se již pojem metrika neužívá [Vaníček 2004].

3.1.4 ISO/IEC 15939

Tato norma má obecnější platnost. Netýká se pouze jakosti, ale obecněji měření všech vlastností (atributů) softwaru a systému. Byla (s předponou ČSN) přijata překladem i jako česká norma. Má název [Vaníček 2004a]:

ISO/IEC 15939 Informační technologie – Softwarové inženýrství – Proces měření softwaru [*Software measurement process*].

Jde o poměrně „mladou“ normu, přijatou mezinárodně v roce 2002. Zavádí přesnou terminologii do oblasti měření a popisuje jednotlivá pravidla a nutné aktivity, které je třeba při měření zajistit, a požadavky na to, aby měření bylo korektní. Je orientována nejen na měření atributů softwaru, ale i na měření atributů informačních systémů, do kterých je software integrován. Protože je pozdějšího data, normy uvedené v předchozích odstavcích ji nerespektují (viz například užívání nejasného pojmu „metrika“, který tato norma nezná). Tato norma bude ovšem podstatným východiskem pro přípravu nové řady ISO/IEC 250xx norem na jakost produktu [Vaníček 2008b].

3.2 Perspektivy mezinárodních norem

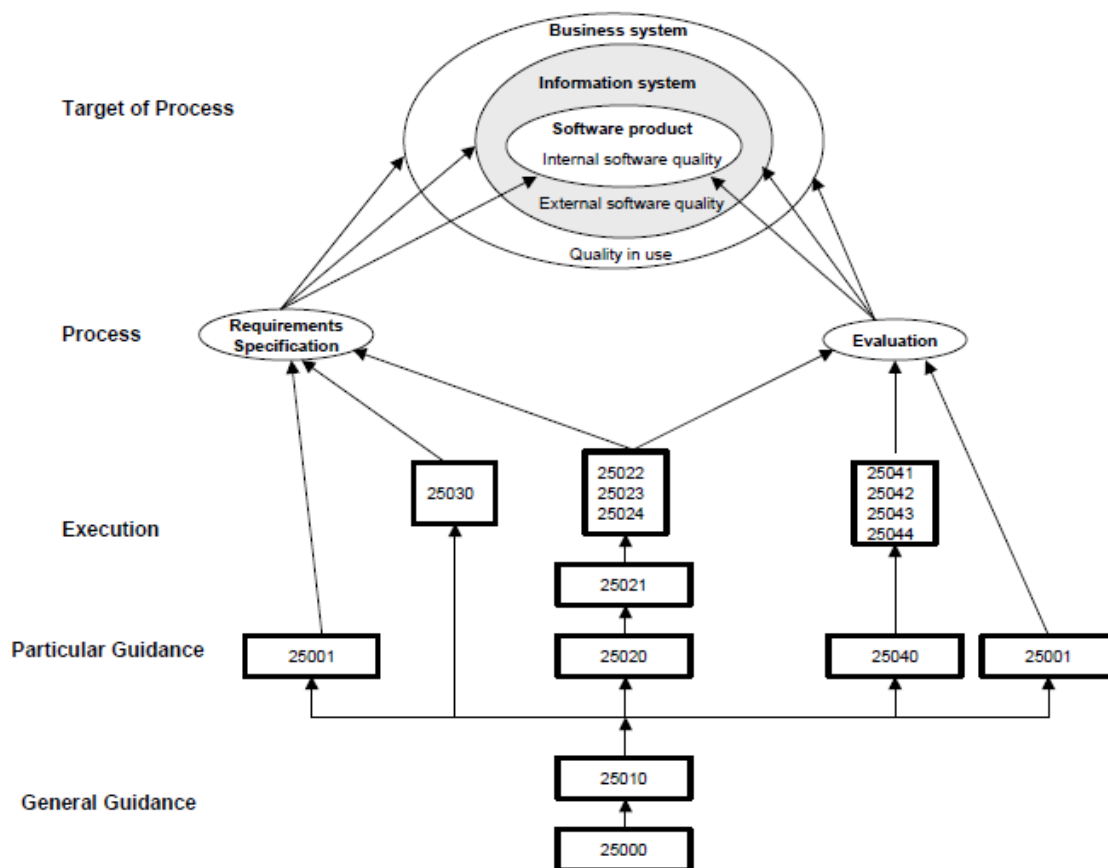
V roce 2000 bylo rozhodnuto [Vaníček 2008a] postupně tyto (výše uvedené) normy nahradit ucelenou soustavou norem řady ISO/IEC 250xx pro kvalitu softwarového produktu. Tato řada je vytvářena mezinárodním týmem řešícím projekt **SQuaRE** (*Software Quality Requirements and Evaluation*). Cílem práce na projektu SQuaRE je nahradit stávající normy pro kvalitu softwaru konsistentní soustavou norem, která by umožnila kvalitu objektivně hodnotit a byla použitelná pro co nejširší typy software.

3.2.1 Model SQuaRE

Obecným cílem vytvoření souboru norem SQuaRE je přechod na logicky uspořádané, obohacené a jednotné série tří vzájemně se doplňujících procesů: specifikace požadavků, měření a hodnocení. Účelem této normy je pomoci těm, kteří vyvíjejí a nabývají softwarové produkty, se specifikací a požadavky na hodnocení kvality. Tato směrnice vytváří kritéria pro stanovení požadavků na kvalitu software výrobků a jejich hodnocení. Model kvality zahrnuje dvě části, sblížit zákaznickovu definici kvality a atributy procesu vývoje. Kromě toho poskytuje řadu doporučených atributů měření kvality software výrobků, které lze použít vývojáři, nabyvateli a hodnotiteli. Je nutno zdůraznit, že řada norem SQuaRE je věnována pouze kvalitě software produktů. SQuaRE ISO/IEC 25000n - Quality Management divize se zabývá specifikací

požadavků na softwarový produkt, měřením a hodnocením, a je oddělená a odlišná od "Quality Management" (řízení kvality) procesů, které je definováno v rodině norem ISO 9000 [ISO/IEC 2004].

Na následujícím obrázku je znázorněn Obecný referenční model SQuaRE. Obecný referenční model SQuaRE byl vytvořen s cílem pomoci uživatelům pohybovat se v řadě norem SQuaRE. Volba příslušných norem a dokumentů z SQuaRE série závisí na roli uživatele a informační potřebě. Je doporučeno, aby všichni uživatelé nejprve konzultovali obecné pokyny (ISO/IEC 25000) spolu s částmi, které jsou relevantní pro jejich konkrétní informační potřeby a roli [ISO/IEC 2004].



Obrázek 4: Obecný referenční model SQuaRE. Zdroj[ISO/IEC 2004].

Řada norem SQuaRE je určena pro vývojáře, nabyvatele a nezávislé hodnotitele softwarových produktů, a zejména osoby odpovědné za definování požadavků kvality softwaru a hodnocení softwarových produktů. Doporučuje se, aby uživatelé řady norem SQuaRE, stejně jako ISO/IEC 14598 a norem řady 9126, také používali tuto mezinárodní normu jako návod k provedení svých úkolů. Hlavní výhody SQuaRE série oproti předešlým standardům zahrnují [ISO/IEC 2004]:

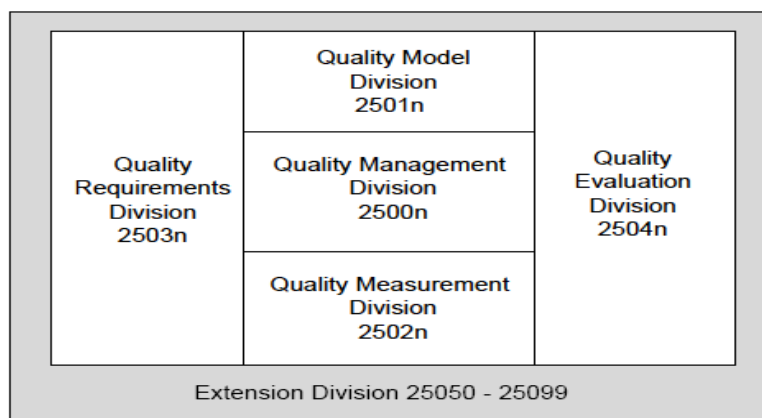
- koordinace vedení měření a vyhodnocení kvality softwarového produktu,
- pokyny pro specifikaci požadavků na kvalitu software výrobků a
- harmonizace s normou ISO/IEC 15939 v podobě referenčního modelu měření kvality software produktu uvedeného v ISO/IEC 25020 - Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) Measurement reference model and guide.

Hlavní rozdíly mezi ISO/IEC 9126, ISO/IEC 14598 a řadou norem SQuaRE dle [ISO/IEC 2004] jsou:

- zavedení nového obecného referenčního modelu,
- zavedení specializovaných, podrobných návodů pro jednotlivé divize,
- zavedení Primitiv pro měření (*Measurement Primitives*) v Quality Measurement Division,
- zavedení Quality Requirements Division,
- začlenění a revize hodnocení procesů,
- zavedení poučení z praktického využití ve formě příkladů,
- koordinace a harmonizace obsahu s normou ISO/IEC 15939.

Struktura norem SQuaRE je dle [ISO/IEC 2009a] členěna do následujících divizí:

- ISO/IEC 2500n - Quality Management Division,
- ISO/IEC 2501n - Quality Model Division,
- ISO/IEC 2502n - Quality Measurement Division,
- ISO/IEC 2503n - Quality Requirements Division,
- ISO/IEC 2504n - Quality Evaluation Division.

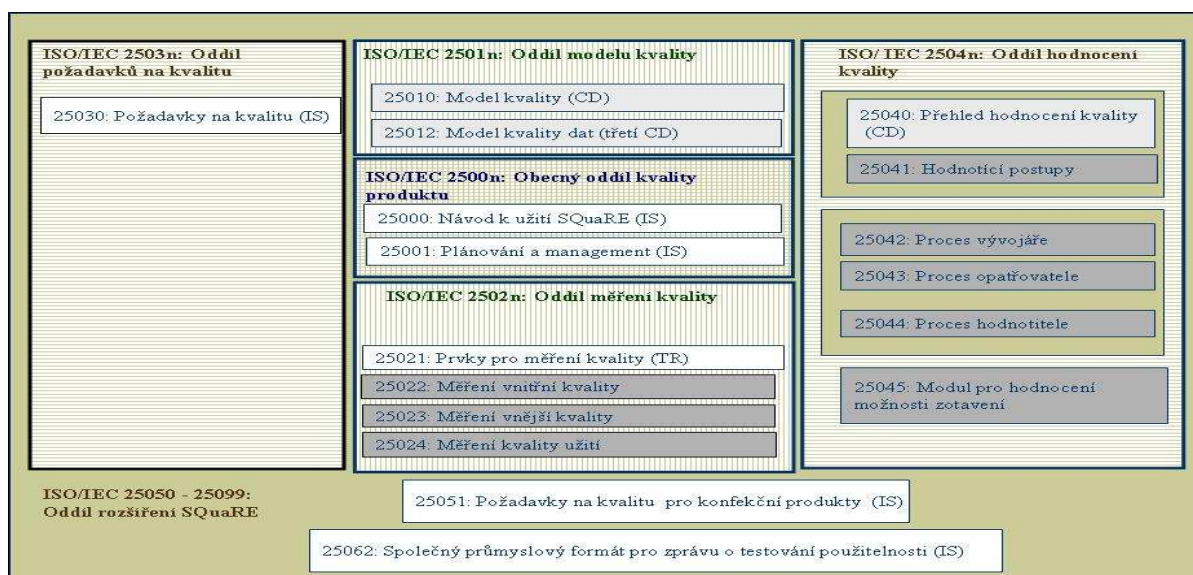


Obrázek 5: Organizace mezinárodních norem SQuaRE. Zdroj [ISO/IEC 2009a.].

Skladba předpokládané řady norem je znázorněna na obrázku 5, na kterém jsou z terminologických důvodů ponechány originální anglické názvy dokumentů. Toto ustanovení představuje přehled o struktuře a obsahu řady norem SQuaRE. Cílem je poskytnout uživatelům této řady norem potřebné informace, které umožní efektivní výběr platných dokumentů.

SQuaRE tedy zahrnuje mezinárodní normy pro kvalitu modelu a měření, stejně jako požadavky na kvalitu a hodnocení a nabízí [ISO/IEC 2004]:

- termíny a definice,
- referenční modely,
- obecný návod,
- průvodce jednotlivých divizí, a
- normy pro specifikaci požadavků, plánování a řízení, měření a hodnocení.



Obrázek 6: Přehled stavu projektu SQuaRE. Zdroj [Vaniček 2008].

Řešení projektu průběžně probíhá [Vaniček 2008a]. Bylo rozhodnuto, že pro číslování norem pro kvalitu softwarových produktů bude užit interval 25000 – 25099. Předpokládané schéma norem této řady je na obrázku č. 4. Na obrázku jsou uvedeny již schválené a vydané normy v bílém poli, normy tučně, normy ve stádiu schvalování v poli světle šedém běžným typem písma a normy, na kterých práce dosud nezačaly, v poli sytě šedém, kurzívou. Charakteristické pro přípravu norem v projektu SQuaRE je uplatnění principu shora – dolů, od obecného k detailům, osvědčeného v systémovém přístupu a softwarových paradigmatech. To vedlo k tomu, že obecné zastřešující normy typu návodů, jak s normami pracovat, jsou dokončeny, normy obsahující konkrétní

podrobnosti, především atributy důležité pro kvalitu a jejich míry však dosud zdaleka ne.

3.2.2 ISO/IEC 2500n - Quality Management Division

Tento dokument [ISO/IEC 2004] obsahuje pokyny pro použití nové řady norem nazývané Software product Quality Requirements and Evaluation (SQuaRE). Účelem této příručky je poskytnout obecný přehled o obsahu SQuaRE, společné referenční modely a definice, stejně jako vztahy mezi dokumenty a umožňuje uživatelům průvodce účinné využití těchto řad norem podle jejich účelu. Tento dokument obsahuje vysvětlení procesu přechodu mezi původní ISO/IEC 9126 a 14598 sérií a SQuaRE, a rovněž informace o tom, jak ISO/IEC 9126 a 14598 série v jejich původní podobě použít.

Standardy, které tvoří tuto divizi, definují všechny běžné modely, termíny a definice uvedené dále ve všech ostatních normách SQuaRE série. Odkaz (průvodce přes SQuaRE dokumenty) a vysoká úroveň praktických návrhů při uplatňování příslušných norem na konkrétní případy použití nabízejí pomoc pro všechny typy uživatelů. Divize poskytuje také požadavky a doporučení pro podporu funkce, která je odpovědná za řízení specifikace požadavků a hodnocení softwarového produktu.

SQuaRE: Přehled dokumentů v rámci edice. SQuaRE řada norem se skládá z následujících dokumentů, které jsou seskupeny do pěti Divizí v rámci modelu SQuaRE. Toto ustanovení představuje stručný přehled všech dokumentů s jejich zařazením do Divize. Tyto dokumenty obsahují předpisy z dokumentů uvedených v závorkách:

- **ISO/IEC 2500n - Quality Management Division**

25000 – Průvodce po SQuaRE: Poskytuje architektonický model, terminologii, přehled dokumentů, které jsou určeny uživatelům a související části série, stejně jako referenční model (ISO/IEC 9126-1 a 14598-1).

25001 – Plánování a management: Poskytuje požadavky a pokyny pro podporu funkce, která je odpovědná za řízení software specifikace požadavků na produkt a hodnocení. (ISO/IEC 14598-2).

- **ISO/IEC 2501n - Quality Model Division**

25010 – Model jakosti: popisuje model interní a externí kvality software pro softwarový produkt a jeho používání. Tento dokument představuje charakteristiky

a podcharakteristiky pro interní a externí kvalitu software a charakteristiky i pro jakost užití (ISO/IEC 9126-1 a 14598-1).

- **ISO/IEC 2502n - Quality Measurement Division**

25020 – Referenční model a průvodce měřením: představuje úvodní vysvětlení, referenční model a definice, které jsou společné pro měření primitiv, vnitřní míry, vnější míry a měření jakosti užití. Obsahuje požadavky, doporučení a rady uživatelům pro výběr a rozvoj a provádění měření z mezinárodních norem (ISO/IEC 9126-1, 9126-2, 9126-3, 9126-4 a 14598-1).

25021 – Měření primitiv: definice a specifikace souboru doporučené základny a odvozených měř, které jsou určeny k použití v průběhu celého životního cyklu vývoje softwaru. Dokument popisuje soubor měř, které lze použít jako vstup pro vnitřní kvalitu software, externí kvalitu software a měření jakosti užití (ISO/IEC 9126-1, 9126-2, 9126-3, 9126-4 a 14598 - 1).

25022 – Měření vnitřní kvality software: definuje vnitřní míry pro kvantitativní měření vnitřní kvality softwaru z hlediska charakteristik a podcharakteristik (ISO/IEC 9126-3).

25023 – Měření vnější kvality software: definuje externí míry pro kvantitativní měření externí kvality software z hlediska charakteristik a podcharakteristik (ISO/IEC 9126-2).

25024 – Měření jakosti užití: popisuje soubor měř pro měření kvality užití. Poskytuje návod pro použití měř jakosti užití (ISO/IEC 9126-4).

- **ISO/IEC 2503n - Quality Requirements Division**

25030 – Požadavky na jakost: obsahuje požadavky a doporučení pro proces používaný pro vývoj požadavků na kvalitu, stejně jako požadavky a doporučení pro požadavky na kvalitu (9126-1, 9126-2, 9126-3, 9126-4, 14598-1, 14598 - 3, 14598-4 a 14598-5).

- **ISO/IEC 2504n - Quality Evaluation Division**

25040 – Referenční model a průvodce hodnocení: obsahuje obecné požadavky na specifikaci a hodnocení kvality software a vysvětluje obecné pojmy. Poskytuje

rámec pro hodnocení kvality softwarových produktů a stanoví požadavky na metody měření a hodnocení softwarového produktu (ISO/IEC 9126-1 a 14598-1).

25041 – Hodnotící moduly: definuje strukturu a obsah dokumentace, která se používá k popisu hodnotícího modulu (ISO/IEC 14598-6).

25042 – Proces hodnocení pro vývojáře: obsahuje požadavky a doporučení pro praktickou implementaci hodnocení softwarového produktu, pokud je hodnocení prováděno souběžně s vývojem (ISO/IEC 14598-3).

25043 – Proces hodnocení pro nabyvatele: obsahuje požadavky, doporučení a pokyny pro systematické měření, posuzování a hodnocení kvality softwarového produktu při získávání "off-the-shelf" softwarových produktů, customizovaných softwarových produktů, nebo modifikaci stávajících softwarových produktů (ISO/IEC 12119 a 14598-4).

25044 – Proces hodnocení pro hodnotitele: obsahuje požadavky a doporučení pro praktickou implementaci hodnocení softwarového produktu, kdy několik stran musí pochopit, přijmout a věřit výsledkům hodnocení (ISO/IEC 14598-5).

3.2.3 ISO/IEC 2501n – Quality Model Division

Normy [ISO/IEC 2009a], které tvoří tuto divizi, prezentují detailní model kvality, včetně charakteristik pro vnitřní a vnější jakost a jakost užití. Kromě toho jsou vnitřní a vnější znaky jakosti softwaru členěny na podcharakteristiky [*subcharacteristics*]. Praktický návod na použití modelu kvality je také k dispozici. Tato mezinárodní norma definuje:

a) model jakosti užití složený z pěti charakteristik (z nichž některé jsou dále rozděleny na podcharakteristiky), které se vztahují na výsledek interakce, kdy je výrobek používán v určitém kontextu užití. Tento model je použitelný pro kompletní lidský-počítačový systém, zahrnující jak v praxi užívané počítačové systémy, tak užívané softwarové produkty.

b) model jakosti produktu se skládá z osmi charakteristik (které se dále dělí na podcharakteristiky), které se vztahují k statickým vlastnostem software a dynamickým vlastnostem počítačového systému. Tento model je použitelný jak pro počítačové systémy, tak pro softwarové produkty.

Charakteristiky definované pro oba modely jsou relevantní pro všechny softwarové produkty a počítačové systémy. Charakteristiky a podcharakteristiky poskytují jednotnou terminologii pro specifikaci, měření a hodnocení systému a jakost softwarových produktů. Poskytují také sadu charakteristik jakosti, proti kterým uvádí požadavky na jakost, které mohou být pro úplnost porovnány.

Rozsah uplatnění modelů kvality obsahuje podporu specifikace a hodnocení software a náročných počítačových systémů z různých úhlů pohledu souvisejících s jejich pořízením, požadavky, vývojem, použitím, hodnocením, podporou, údržbou, zajištěním kvality a kontroly, a auditu. Tyto modely mohou být používány například vývojáři, nabyvateli, zaměstnanci zajištění kvality a kontroly a nezávislymi hodnotiteli, zejména těmi, kteří jsou odpovědní za specifikaci a hodnocení kvality produktů software. Aktivity v průběhu vývoje produktu, které mohou těžit z použití modelů kvality, zahrnují:

- identifikaci softwaru a systémových požadavků,
- ověření komplexnosti definice požadavků,
- identifikaci cílů návrhu software a systému,
- identifikaci cílů testování software a systému,
- identifikaci kritérií kontroly jakosti jako součásti systému zabezpečování jakosti,
- identifikaci akceptačních kritérií pro softwarový produkt a/nebo softwarově náročných počítačových systémů,
- stanovení měření charakteristik jakosti na podporu těchto aktivit.

3.2.4 ISO/IEC 2502n – Quality Measurement Division

Normy [ISO/IEC 2004], které tvoří tuto divizi, zahrnují referenční model měření kvality softwarového produktu, matematické definice měření jakosti, a praktické návody pro jejich použití. Prezentovaná měření se vztahují na vnitřní jakost software, externí jakost software a jakost užití. Definovány a prezentovány jsou prvky měření kvality (QME), které jsou základem pro tato měření.

Asi nejdůležitější mezinárodní normou pro vznik této práce je [ISO/IEC 2011a]. Účelem tohoto dokumentu je definovat a/nebo navrhnout výchozí množinu prvků měření kvality (QME), které mají být použity během životního cyklu produktu pro hodnocení jakosti a požadavků na jakost systémů a softwarových produktů (SQuARE). Dokument také uvádí soubor pravidel pro návrh QME nebo ověřování designu

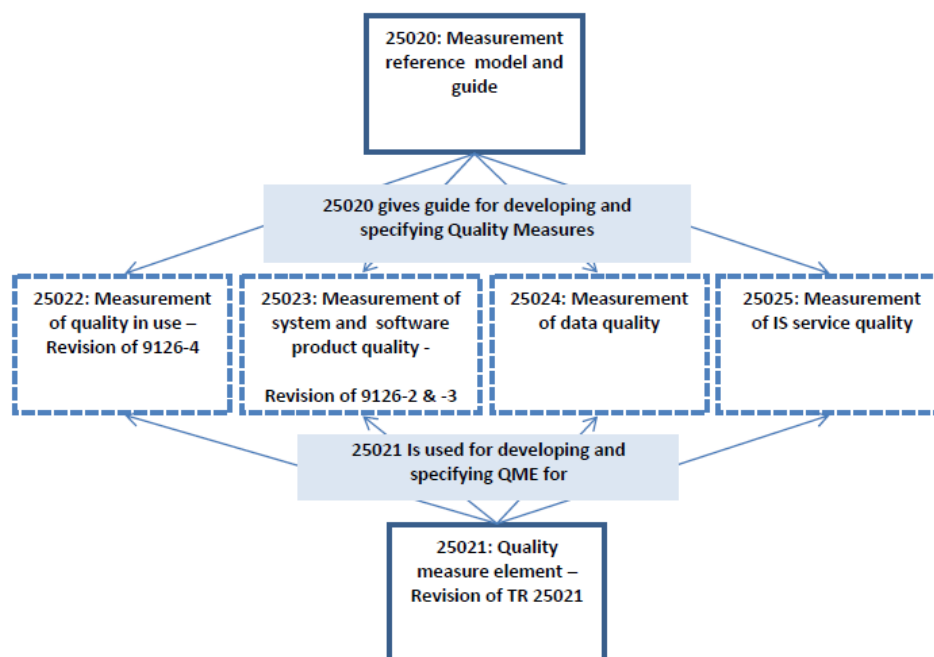
existujících QME. Obsah tohoto dokumentu vytváří vazbu mezi řadou norem ISO/IEC 9126 a následnou řadou norem SQuaRE. Prvkům měření kvality je věnována zvláštní kapitola (kap. 6) této práce viz dále.

Počet QME pro měření kvality, které kvantifikují některé charakteristiky a sub-charakteristiky, představuje základní seznam, který se používá při konstrukci měření jakosti, jak je uvedeno v ISO/IEC TR 9126-2, ISO/IEC TR 9126 -3 a ISO/IEC TR 9126-4. Měření jakosti uvedené v sérii SQuaRE vychází z řady ISO/IEC TR 9126, ale nejde pouze o jediný zdroj. Při posuzování vybraných měření kvality, musí uživatel nejprve pochopit definice každé z vlastností v souvislosti s QME využitých pro měření kvality.

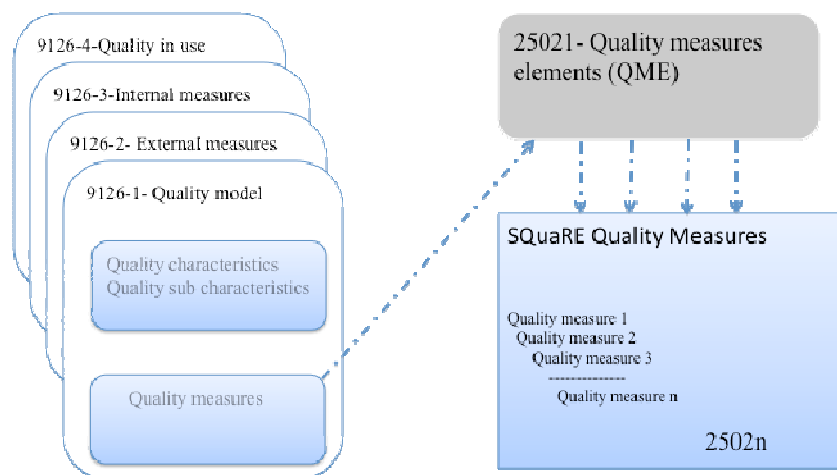
Hlavním účelem vymezení a používání prvků měření kvality v tomto dokumentu je:

- poskytnout návod pro organizace, vytváření a provádění svých vlastních QME,
- zajištění jednotného používání specifických QME pro měření produktu,
- pomoci identifikovat unikátní soubor QME, který je požadovaný pro odvození všech měření kvality daných charakteristik nebo podcharakteristik produktu.

QME je běžnou součástí celé řady měření jakosti. Zamýšlené použití tohoto dokumentu je, že si uživatelé budou moci vybrat z platných QME pro definování měření vnitřní jakosti, vnější jakosti, měření kvality dat nebo jakosti užití. Tyto QME je možné použít pro stanovení požadavků na jakost, hodnocení kvality produktu, ale není nutné omezit se pouze na ty. Doporučuje se proto používat tento dokument před nebo ve spojení s řadou norem ISO/IEC 2502n.



Obrázek 7: Struktura Quality Model Division. Zdroj [ISO/IEC 2011a].



Obrázek 8: Vztah ISO/IEC 25021 jako spojitost mezi sérií 9126 a řadou norem SQaRE Zdroj [ISO/IEC 2011a].

Mezinárodní norma série ISO/IEC 9126 se skládá ze čtyř dokumentů, které uvádí a popisují sadu charakteristik jakosti, podcharakteristik a metrik kvality (měření), které se vztahují k modelu kvality. ISO/IEC 25010 SQaRE zařazuje kvalitu produktu do charakteristik, které se dále dělí na podcharakteristiky a vlastnosti jakosti. Pro každé měření v rámci ISO/IEC 9126 série jsou nejméně dva QME. Vlastnosti (výrobku), jsou spojeny s QME (ISO/IEC 25020), za použití metody měření. Série 2502n navrhuje a popisuje QME a měření jakosti pro model kvality.

Tato mezinárodní norma obsahuje následující informace:

- Zajistit požadavky na vymezení QME na základě požadavků na jakost produktu s příklady,
- poskytnout prvotní sadu QME,
- poskytnout návod pro použití metody měření, definovat a kvantifikovat vlastnosti produktu (cílové entity) pro QME.

Pokud uživatelé definují měření kvality pro daný produkt, musí specifikovat jednotlivé QME podle metody měření uvedené v tabulce 2. To samé by mělo být použito pro úpravu QME. Uživatel této normy by měl přijmout QME na základě požadavků na kvalitu produktu a kritérií hodnocení.

3.2.5 ISO/IEC 2503n – Quality Requirements Division

Standardy [ISO/IEC 2006a], které tvoří tuto divizi, pomáhají specifikovat požadavky na kvalitu. Tyto požadavky na kvalitu mohou být použity v procesu získání požadavků na kvalitu pro vývoj softwarového produktu, nebo jako vstupy pro proces hodnocení.

Je důležité identifikovat a specifikovat požadavky na kvalitu software jako součást upřesnění požadavků pro softwarový produkt. Software je obvykle součástí většího systému. Požadavky na systém a požadavky na software jsou úzce spojené, a proto nelze na požadavky na software nahlížet izolovaně. Tato mezinárodní norma se zaměřuje na požadavky kvality software, ale bere v úvahu systémové hledisko. Požadavky na kvalitu softwaru lze rozdělit pomocí modelu kvality, např. modelu kvality definovaného v ISO/IEC 9126-1 [ISO/IEC 25010]. Míry atributů těchto charakteristik a jejich podcharakteristik mohou být používány k určení požadavků na jakost softwaru a hodnocení kvality softwarového produktu.

Požadavky na jakost software se zabývají řešením důležitých otázek jakosti softwarových produktů. Požadavky na jakost softwarových produktů jsou potřebné pro:

- specifikace (včetně smluvního ujednání a výzvy k podání nabídek),
- plánování (analýza proveditelnosti a překlad vnějších požadavků na kvalitu softwaru na interní požadavky na kvalitu software),
- vývoj (včasná identifikace problémů s kvalitou během vývoje),
- hodnocení (objektivní hodnocení a certifikace kvality softwarového produktu).

Pokud požadavky na kvalitu software nejsou jasně a jednoznačně uvedeny, mohou být viděny, interpretovány, realizovány a vyhodnocovány rozdílně různými lidmi. To může

mít za následek: nekvalitní software, který je v rozporu s požadavky uživatelů; uživatele, klienty a vývojáře, kteří jsou nespokojeni; čas a překročení nákladů na přepracování software.

Použití této mezinárodní normy by mělo pomoci zajistit, aby požadavky na jakost software byly:

- v souladu s potřebami zainteresovaných stran,
- uvedené jednoznačně a přesně,
- správné, úplné a konzistentní,
- ověřitelné a měřitelné.

3.2.6 ISO/IEC 2504n – Quality Evaluation Division

Standardy [ISO/IEC 2004], které tvoří tuto divizi, poskytují požadavky, doporučení a pokyny pro hodnocení softwarového produktu, ať už provedeného hodnotitelem, nabyvatelem, nebo vývojářem. Podpora pro zdokumentování měření jako hodnotícího modulu je také prezentována.

Hodnocení je dle [ISO/IEC 2011b] systematické určování rozsahu, do jaké míry subjekt splňuje svá stanovená kritéria. Hodnocení kvality produktů je důležité jak pro nabytí, tak pro vývoj software. Relativní význam různých charakteristik jakosti softwaru závisí na zamýšleném použití nebo cílech systému, jehož součástí je hodnocený software; produkty musí být hodnoceny, aby bylo možné rozhodnout, zda příslušné charakteristiky jakosti splňují požadavky na systém.

Tento dokument je součástí ISO/IEC 250nn série standardů SQuaRE. ISO/IEC 25040 obsahuje všeobecné požadavky a doporučení pro hodnocení kvality produktů, stejně jako související obecné koncepty. Tento dokument obsahuje specifické otázky týkající se vývojářů, nabyvatelů a nezávislých hodnotitelů na základě ISO/IEC 25040.

Tato mezinárodní norma obsahuje požadavky, doporučení a pokyny pro hodnocení kvality produktů speciálně pro vývojáře, nabyvatele a nezávislé hodnotitele. Tato norma není omezena na žádnou konkrétní oblast použití a mohou být použity pro hodnocení kvality všech typů produktů.

Tento dokument obsahuje popis procesu hodnocení kvality produktů a uvádí specifické požadavky pro použití tohoto procesu z pohledu developera, nabyvatele, a nezávislého hodnotitele. Proces hodnocení je možné použít pro různé účely a přístupy.

Tento proces může být použit pro hodnocení jakosti „pre-developed“ software, komerční „off-the shelf“ software nebo software na zakázku, a může být použit v průběhu, nebo po procesu vývoje.

Tato mezinárodní norma je určena pro ty, kteří jsou odpovědní za hodnocení produktu a je vhodná pro vývojáře, nabyvatele a nezávislé hodnotitele softwarových produktů. Popisy v této normě jsou založeny hlavně na popisech specifických norem ISO/IEC 14598-3, -4, -5 a, které budou nahrazeny tímto standardem. Tato mezinárodní norma není určena pro hodnocení dalších aspektů produktu (jako jsou funkční požadavky, procesní požadavky, obchodní podmínky atd.).

Dále je popsána koncepce hodnocení z hlediska jednotlivých rolí (rámec hodnocení jakosti produktů z hlediska každé role) [ISO/IEC 2011b].

Rozdíl mezi vývojáři, nabyvateli a nezávislými hodnotiteli se určuje na konkrétní činnosti hodnocení z pohledu odpovědnosti každé role a kategorie hodnocení cílové entity. Zjednodušeně řečeno, proces hodnocení jakosti produktů u jednotlivých rolí je stejný, ale cílový subjekt hodnocení je rozdílný mezi účely hodnocení, které závisejí na požadovaných potřebách.

Obr. 9 ukazuje celkový rámec hodnocení jakosti produktů. Hodnocení jakosti produktu je považováno za systém, který je složen ze vstupů, výstupů, omezení a zdrojů procesu hodnocení. Vstupy, výstupy, omezení a zdroje procesu hodnocení jsou různé pro jednotlivé role a účel hodnocení.



Obrázek 9: Celkový rámec hodnocení jakosti produktů. Zdroj [ISO/IEC 2011b].

3.2.7 ISO/IEC 25050 – 25099 SQuaRE Extension Division

Ty standardy [ISO/IEC 2004] v současné době zahrnují i požadavky na kvalitu komerčního standardního software a běžných průmyslových formátů pro použitelnost

zprávy. Komerční Off-The-Shelf (COTS) softwarové produkty jsou používány ve stále více nejrůznějších aplikačních oblastí a jejich správné fungování je často zásadní pro podnikání, bezpečnost nebo osobní aplikace.

3.2.8 Nedostatky modelu SQuaRE

Přestože záměr a cíl vytvořit novou jednotnou řadu norem, dodržovat jednotnou terminologii, odstranit redundanci a opakování stejných informací v jednotlivých dokumentech řady, která nutně vede k vzájemné nekonzistenci norem, a vyvarovat se všem nejasným formulacím, se zdá být zcela zřejmý, realizaci tohoto projektu provází různé problémy.

Jedním z problémů, na které autor při studiu standardů narazil, je jazyk těchto norem. Relativně často se vyskytují věty a formulace, které jsou pro běžného čtenáře velmi těžko čitelné a často je obtížné extrahovat význam sdělení. Rovněž kriticky lze v současné době hodnotit cíl, který sleduje projekt SQuaRE z pohledu závěrečného hodnocení softwarového produktu. Z dostupných pramenů není dosud zcela jasná metodika, jak lze jednotlivé výsledky dílčích měření shrnout do jednoho závěrečného hodnocení.

Za nejpodstatnější překážky na cestě k vytvoření skutečně použitelné a přitom jednoduché soustavy norem jsou dle Vaníčka [Vaníček 2004b] následující dva hlavní problémy:

1. Formulace požadavků na jakost. Tato část přípravy norem řady 250xx je patrně jednou z nejslabších. Převod mezi potřebami a požadavky na jakost je neřešený problém. Zpravidla jej není schopen budoucí uživatel systému provést. Často není schopen jej ani sledovat a posoudit. Většinou jej za uživatele dělá budoucí dodavatel. Zde je třeba si uvědomit, že lidé i společnosti obvykle nemají názory. Lidé mívají zájmy. Že zájem dodavatele nemusí být shodný s potřebami uživatele, je jasné. Pokud nebudou dávat normy návod, jak požadavky na základě potřeb vymezit a jak je objektivně kvantifikovat, bude význam hodnocení jakosti mírou splnění těchto požadavků vždy velmi problematický a víceméně subjektivní.

2. Výběr vhodných atributů a jejich měř. Zdá se, že největší nedostatek současného stavu norem pro jakost produktů se v rámci projektu SQuaRE zatím řešit nedaří. Úvahy o tom, které konkrétní vnější atributy popisují dobře jednotlivé charakteristiky jakosti a které vnitřní atributy jsou jejich vhodnými prediktory, vážnou.

Bez shody na tomto výběru nebo asi lépe bez shody na několika sadách takovýchto výběrů pro různé typy informačních systémů a různá paradigmatata jejich vývoje bude význam všech dalších norem jen velmi omezený. Teprve na základě takového výběru lze definovat primitiva pro měření a poté příslušné míry.

Jako další z řady problémů vidí Vaníček [Vaníček 2008c]:

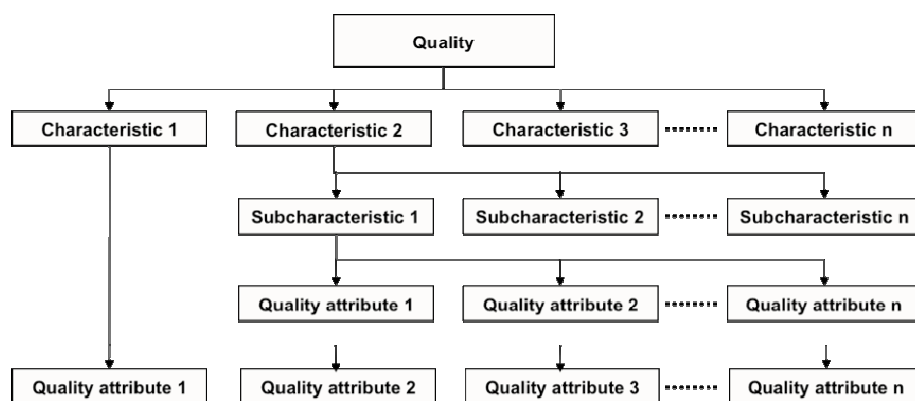
- mezinárodní řešitelský tým projektu je nehomogenní,
- prodlužování a časová náročnost řešení,
- nárůst počtu dokumentů a složitost systému norem, které mohou odradit uživatele,
- někdy nejasné definování pojmů.

S tím nelze než souhlasit. I přes tento kritický pohled autor považuje úkol vytvořit konsistentní a snadno použitelnou soustavu norem pro hodnocení kvality informačních systémů a softwarových produktů za velmi důležitý a již dnes je možné z dílčích výsledků projektu čerpat informace.

3.3 Charakteristiky jakosti

Kvalita systému je dle [ISO/IEC 2009a], do jaké míry tento systém splňuje stanovené a předpokládané potřeby svých různých zainteresovaných stran, a tím poskytuje hodnotu. Tyto stanovené a předpokládané potřeby jsou zastoupeny v řadě norem modelu jakosti SQuaRE, které kategorizují kvalitu výrobku do charakteristiky, která se v některých případech dále rozdělí do podcharakteristik [*subcharacteristics*]. Tato hierarchická dekompozice nabízí pohodlné členění kvality produktu. Charakteristiky jakosti, podcharakteristiky nebo další členění lze měřit. Nejnižší úroveň se skládá z atributů (obr. 10). Atributy jakosti lze měřit dle měření kvality prvků (ISO/IEC 25021).

Charakteristiky jakosti definují podcharakteristiky, a standard umožňuje uživatelům definovat sub-sub-charakteristiky v hierarchické struktuře. Definované charakteristiky jakosti zahrnují všechny aspekty jakosti zájmu pro většinu softwarových produktů, a jako takové mohou být použity jako kontrolní seznam pro zajištění úplného pokrytí kvality [ISO/IEC 2006a].

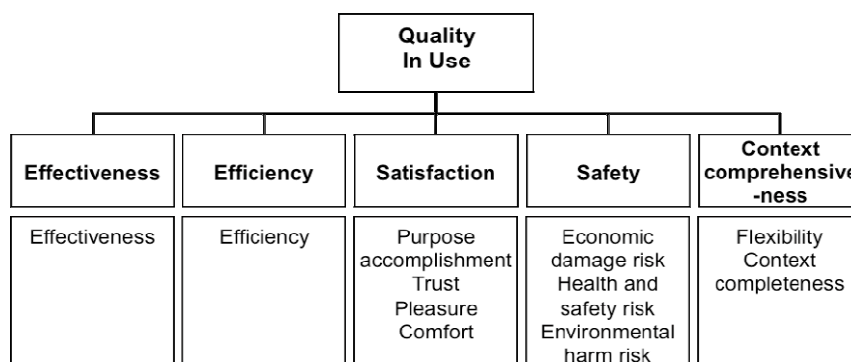


Obrázek 10: Struktura modelu kvality. Zdroj [ISO/IEC 2009a].

V současné době existují 3 modely kvality v SQuaRE: model kvality užití, model jakosti výrobků software v této mezinárodní normě a model kvality dat v ISO/IEC 25012. Modely kvality dohromady slouží jako rámec, který zajistí, že všechny znaky jakosti jsou uvažovány. Tyto modely poskytují soubor charakteristik významný pro široký okruh zúčastněných stran: například vývojáři softwaru, systémoví integrátoři, nabyvatelé, majitelé, správci, dodavatelé a koneční uživatelé.

3.3.1 Jakost užití

Model kvality užití [ISO/IEC 2009a] definuje pět vlastností vztahující se k výsledkům interakce se systémem: efektivita, účinnost, spokojenost, bezpečnost a kontext úplnosti. Kvality použití systému charakterizuje dopad, který má výrobek (systém nebo softwarový produkt) na zúčastněné strany. Je určen kvalitou software, hardware a prostředím, charakteristikou uživatelů, úkoly a sociálním prostředím. Všechny tyto faktory zahrnují kvalitu užití systému.



Obrázek 11: Model kvality užití systému. Zdroj [ISO/IEC 2009a].

Charakteristiky jakosti užití:

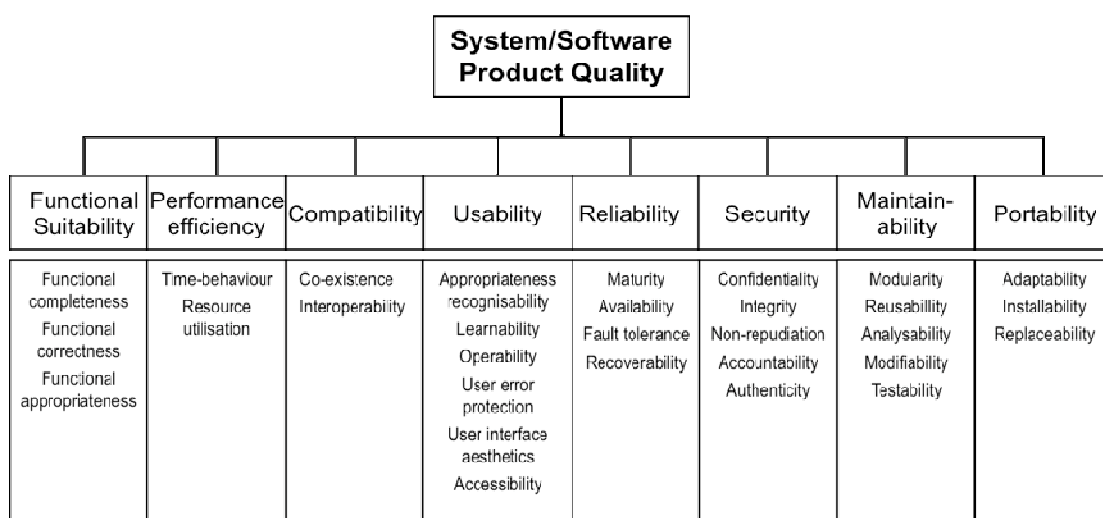
- **Efektivnost** je vymezena jako přesnost a úplnost, s nimiž uživatelé dosahují stanovených cílů.

- **Účinnost** je vymezena jako přesnost a úplnost, s nimiž uživatelé dosažení stanovených cílů.
- **Spokojenost** je vymezena - do jaké míry jsou splněny potřeby zúčastněných stran, pokud je výrobek v určitém kontextu používán.
- **Bezpečnost** je vymezena - do jaké míry výrobek nebo systém za určitých podmínek nevede do stavu, ve kterém je ohroženo zdraví, lidský život, majetek nebo životní prostředí.
- **Úplnost kontextu** je vymezena - do jaké míry výrobek může být použit efektivně, účinně, bezpečně a uspokojivě v obou uvedených kontextech použití a v souvislostech, nad původně označený rámeček.

3.3.2 Kvalita softwarového produktu

Model jakosti softwarového produktu [ISO/IEC 2009a] kvalifikuje atributy kvality systém /software výrobků do osmi charakteristik: funkční vhodnost, účinnost výkon, kompatibilitu, použitelnost, spolehlivost, bezpečnost, udržovatelnost a přenositelnost. Každá charakteristika se skládá ze souboru souvisejících podcharakteristik.

Kvalita systému je výsledkem jakostí systémových prvků a jejich vzájemného působení. Mezinárodní standard [ISO/IEC 2011b] se zaměřuje na kvalitu software jako součást systému. Kvalita softwaru je, jak již bylo řečeno, schopnost softwarového produktu uspokojovat stanovené a předpokládané potřeby, pokud je používán za stanovených podmínek. Model jakosti softwarového produktu, který je uveden v uvedené v ISO/IEC 9126-1 [ISO/IEC 25010], definuje osm charakteristik jakosti (viz obrázek 12).

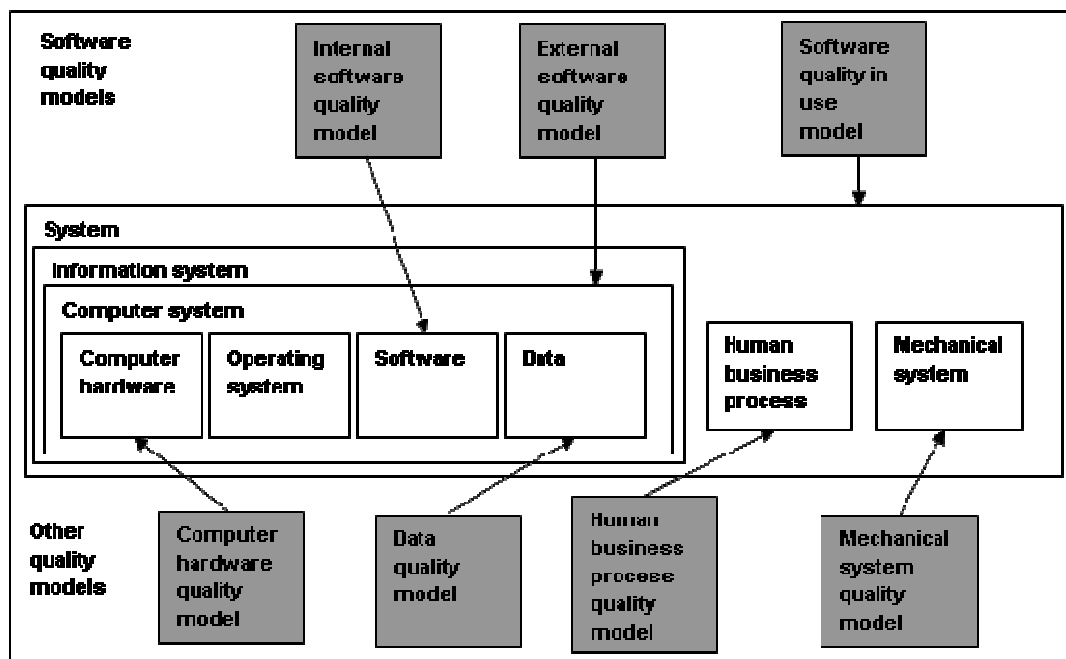


Obrázek 12: Model kvality softwarového produktu. Zdroj [ISO/IEC 2009a].

Definice charakteristik kvality software:

- **Funkční přiměřenost** - do jaké míry výrobek zabezpečuje funkce, které splňují stanovené a předpokládané potřeby, je-li přípravek používán za stanovených podmínek.
- **Účinnost** - výkonnost ve vztahu k výši prostředků používaných za stanovených podmínek.
- **Schopnost spolupráce** - do jaké míry dva (příp. více) systémy nebo jejich součásti mohou vyměňovat informace a/nebo vykonávat své požadované funkce při sdílení stejného hardware nebo software prostředí.
- **Použitelnost** - do jaké míry výrobek může být používán určenými uživateli k dosažení stanovených cílů účinně, efektivně a uspokojivě ve stanoveném kontextu používání.
- **Bezporuchovost** - do jaké míry systém nebo jeho součást plní stanovené úkoly v rámci stanovených podmínek na určitou dobu.
- **Bezpečnost** - do jaké míry jsou informace a data jsou chráněna tak, aby nedošlo k neoprávněné modifikaci nebo přístupu neautorizovanou osobou nebo systémem, a oprávněným osobám nebo systému není odepřen přístup k nim.
- **Udržovatelnost** - stupeň účinnosti a efektivity, s nimiž může být produkt modifikován.
- **Přenositelnost** - do jaké míry systém nebo jeho součást může být účinně a efektivně převedena z jednoho hardware, software nebo jiného prostředí do druhého.

Kvalita software z pohledu použití souvisí dle [ISO/IEC 2006a] s uplatněním software v jeho operačním prostředí, k provádění konkrétních úkolů podle konkrétních uživatelů. Následující obrázek ukazuje interakci mezi různými modely kvality a systémovým modelem. SQUARE série mezinárodních norem pokrývají pouze modely kvality software, nikoli však další modely kvality, které jsou na obrázku.



Obrázek 13: Příklad modelu systému a modelů kvality. Zdroj [ISO/IEC 2006a].

Model kvality slouží jako rámec s cílem zajistit, že všechny aspekty kvality jsou posuzovány z pohledu vnitřní jakosti, vnější jakosti, a jakosti užití.

3.4 Atributy jakosti

Měřitelná fyzická nebo abstraktní vlastnost entity se nazývá **atribut** [atribut] [Vaníček 2004a]. Ideální by bylo, kdyby bylo možné přiřadit každé podcharakteristice jeden nebo několik atributů, změřit je a na základě výsledků měření vyhodnotit úroveň jakosti, které bylo v této podcharakteristice dosaženo. Právě tak, jako rozklad jakosti na její charakteristiky a podcharakteristiky nebyl stromový, nemůže být stromová ani další úroveň, rozklad na jednotlivé atributy. **Jeden atribut zpravidla ovlivňuje více podcharakteristik jakosti, často i podcharakteristik zahrnutých do různých charakteristik jakosti.** Některé dvojice charakteristik jakosti ovlivňuje stejným, jiné dvojice opačným směrem. Z hlediska toho, kdy je možné atributy jakosti hodnotit, je důležité následující dělení [Vaníček 2004a]:

- **Vnější atributy jakosti** [external quality attributes] jsou atributy chování systému, jehož jakost hodnotíme, nebo systému, jehož je hodnocený produkt částí. Jde vlastně o nepřímé měření produktu, pomocí chování systému, do kterého je produkt zařazen.

- **Vnitřní atributy jakosti** [*internal quality attributes*] jsou atributy systému samotného. Mohou být měřeny buď přímo nebo nepřímo, tak, že hodnoty jejich měř získáme výpočtem z hodnot měř jiných přímo měřených atributů. Důležité je, že jde o atributy, které ovlivňují jakost systému, do kterého je produkt zařazen, tedy jeho chování vůči uživateli.
- **Atributy jakosti užití** [*quality in use attributes*] jsou atributy využívání systému v konkrétním nasazení u uživatele. Týkají se vždy jen systému jako celku a jeho konkrétního nasazení. Jsou ovlivněny i jakostí konkrétních uživatelů a je obtížné pomocí nich hodnotit jednotlivé produkty, které jsou v rámci systému užity.

Nelze jednoznačně rozhodnout, kterým atributům dát při hodnocení jakosti přednost. Důležité jsou obě skupiny. Jsou však k dispozici v různých etapách životního cyklu systému a pro různé skupiny lidí, které mají se systémem co do činění. Vnitřní atributy mají význam především pro vývoj systému. Vnější budou zajímat zákazníka a budoucího uživatele. Ten často nebude mít ani přístup k údajům (například k zdrojovým kódům softwaru), které by mu umožnily vnitřní atributy vyhodnotit [Vaníček 2004a].

Vztah mezi vnitřními atributy jakosti a jejími vnějšími atributy není vztahem jedna k jedné. Pro týž vnější atribut lze zpravidla nalézt více vnitřních atributů, pomocí kterých lze míru tohoto atributu předpovědět. Na druhé straně týž vnitřní atribut může indikovat míru několika vnějších atributů. Dokonce může jít i o atributy, které se řadí do různých podcharakteristik nebo dokonce i charakteristik jakosti produktu [Vaníček 2004a].

Důležitou vlastností vnějších i vnitřních atributů jakosti je, že **musí být měřitelné**, tedy popsatelné čísly nebo jinými abstraktními entitami. Vývoj informačních systémů a softwarových produktů je dnes vždy výrazně týmovou prací a je pro něj charakteristické hierarchické skládání celku z komponent. Zvláště pro vnitřní atributy je tedy klíčové mít k dispozici možnost určit míru pro atribut celku z měř zjištěných pro jeho komponenty. Tato možnost je zajištěna pro případ měření v měřicích stupnicích poměrového typu. Podmínky pro možnost takového měření jsou však poměrně náročné a ne vždy je možné je splnit. Pokud máme možnost atributy měřit pouze v měřicí stupnici ordinálního typu, je třeba ověřit aspoň to, že míry jsou vůči skládání komponent v celek monotónní. To znamená, že zmenšení míry komponenty nevede

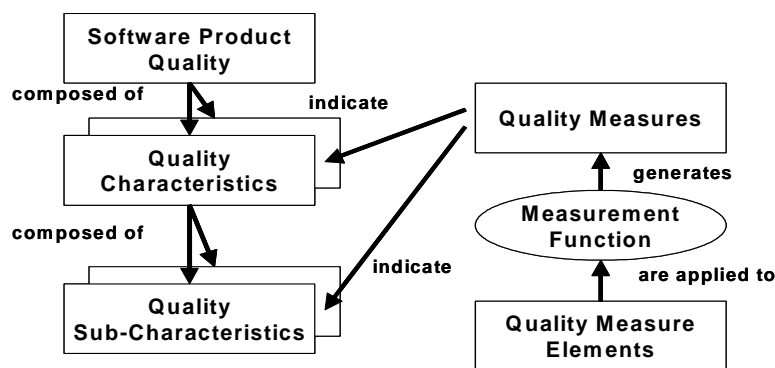
k zvětšení míry celku. Kdyby tento požadavek míra daného atributu nesplňovala, nemělo by smysl snažit se o zlepšování jakosti komponent [Vaníček 2004a].

4 Měření jakosti informačních systémů

Měřením se obvykle rozumí dle [Vaníček 2004a] popis studovaných skutečností čísly. I tak zvané „měkké problémy“, jako je míra uspokojení a „měkké pojmy“, jakým je nepochybně pojem potřeb uživatelů, lze popsat přesně. Jakost je tedy možné hodnotit nejen na základě „mlhavých pocitů“, ale i studiem exaktních hodnot, získaných jako výsledek měření, popisujících výsek světa kolem nás. V některých případech se ovšem ukazuje, že čísla nejsou jediným možným prostředkem pro tento popis a někdy ani prostředkem nejhodnějším.

Měření je cesta, jak objektivizovat jakékoliv hodnocení, tedy i hodnocení jakosti. Jeho podstatou je popis vybraných atributů entit z reálného světa formálními objekty, s kterými může pracovat matematika a výpočetní technika, nejčastěji tedy čísla. Tento popis musí zachovávat vše podstatné, co nás na daném pohledu na daný výsek světa zajímá. Tyto důležité vlastnosti a vztahy lze popsat pomocí relací a operací, které jsou ve své podstatě zvláštním případem relací. Velký význam pro možnost interpretace výsledků získaných výpočty z naměřených hodnot do reálného světa mají měřicí stupnice [Vaníček 2004b].

Měření není vždy určeno jednoznačně. Zpravidla při něm máme více možností, jak popisovaný výsek světa transformovat do čísel nebo zvolených formálních objektů. Je možné například zvolit jinou základní jednotku, se kterou ostatní porovnááme, nebo jiný počátek. Počítat pak budeme pochopitelně s jinými čísly a získáme jiné výsledky při totéž správném postupu. My samozřejmě máme zájem především o poznatky a skutečnosti. Vzniká tedy přirozená otázka, jak přenést získané výsledky zpět z formálního světa do empirického. K tomu účelu musíme získaným formálním objektům (výsledkům výpočtu, či užití dané teorie) přiřadit zpět objekty reálného světa. Tento krok je nazýván interpretace. Při interpretaci však může nastat problém. Měření nebylo jednoznačné. Zvolíme-li jinou možnost formálního popisu skutečnosti, dostaneme jiný výsledek a není automaticky jasné, zda interpretace tohoto jiného výsledku bude o skutečnosti říkat totéž, nebo něco jiného. Pokud by výsledek závisel na svobodné volbě měření, neměly by získané výpovědi o skutečnosti rozumný smysl. Nebylo by je možné relevantně interpretovat. Při interpretaci je proto nutné posoudit, které výsledky platné ve formální struktuře mají v empirické svůj význam, a které nikoliv [Vaníček 2004a].



Obrázek 14: Referenční model měření kvality softwarového produktu. Zdroj [ISO/IEC 2009a].

Na obrázku 14 jsou znázorněny vztahy mezi ISO/IEC 25010 modelem kvality, měřením v ISO/IEC 2502n, a modelem měření navrženém v ISO/IEC 15939. Kvalitativní vlastnosti se měří uplatněním metody měření. Metoda měření je logická posloupnost operací použitá ke kvantifikaci vlastností s ohledem na stanovený rozsah. Výsledkem použití metody měření je prvek metody měření. Charakteristiky a podcharakteristiky jakosti lze kvantifikovat použitím měřicích funkcí. Funkce měření je algoritmus použitý ke sloučení prvků měření kvality. Výsledkem použití funkce měření je míra jakosti software. Tímto způsobem se stává měření jakosti software kvantifikací kvalitativních charakteristik a podcharakteristik. Více než jedna míra jakosti software může být použita k měření jakosti charakteristik nebo podcharakteristik [ISO/IEC 2009a].

4.1 Vlastnosti software

Kapitola Vlastnosti software vychází z mezinárodního standardu [ISO/IEC 2006a]. Některé vlastnosti software jsou inherentní v softwarovém produktu (např. počet řádků kódu, přesnost numerických výpočtů), některé z nich jsou k softwarovému produktu přiřazeny (např. záruka, cena). Kvalita softwarového produktu v určitém kontextu použití je dána jeho vnitřními vlastnostmi.

Inherentní vlastnosti mohou být klasifikovány buď jako funkční vlastnosti, nebo kvalitativní vlastnosti. Funkčních vlastností určují, co software je schopen vykonávat. Kvalitativní vlastnosti určují, jak dobře je software vykonáván. Kvalitativní vlastnosti jsou inherentní v softwarovém produktu a souvisejícím systému. Přiřazené vlastnosti proto nejsou považovány za charakteristiku jakosti software, protože mohou být

změněny, aniž by se změnil software. Obrázek 15 ilustruje tuto klasifikaci vlastností software.

Software properties	Inherent properties	Domain-specific functional properties
		Quality properties (functional suitability, reliability, performance efficiency, operability, security, compatibility, maintainability, portability,)
	Assigned properties	Managerial properties like for example price, delivery date, product future, product supplier

Obrázek 15: Vlastnosti software. Zdroj: [ISO/IEC 2006a].

Pro každou podcharakteristiku, schopnost software, je určena sada statických vnitřních vlastností, které lze měřit. Charakteristiky a podcharakteristiky mohou být měřeny externě v rozsahu schopností, které poskytuje systém obsahující software.

Externí míry jakosti systému/software poskytují "black box" pohled na systém/software a adresují vlastnosti spojené s realizací tohoto software na počítačový hardware a operační systém. Vnitřní míry kvality softwaru poskytují "white box" pohled na software a adresuje statické vlastnosti softwarového produktu, které obvykle mají k dispozici pro hodnocení průběhu vývoje. Kvalita softwaru měřená interně má vliv na kvalitu systému/software měřenou externě, což má zase vliv na jakost užití systému.

Interní míry založené na kontrole statických vlastností mohou být použity k měření vnitřním vlastnostem software produktu (viz tabulka 1). Statická analýza metody zahrnuje kontrolu a automatické analytické nástroje. Produkty zahrnují požadavky a konstrukční dokumentace, zdrojový kód, a zkušební postupy. Externí míry dynamických vlastností lze použít k měření inherentních vlastností počítačového systému, a vlastností systémových závislostí softwarového produktu.

Míry jakosti užití (z testování a pozorování výsledků skutečné nebo simulované použití) měří vnitřní vlastnosti systému, který může zahrnovat software, hardware, komunikaci a uživatele, a vlastnosti software závislé na systému - náročný počítačový systém nebo softwarový produkt. Míry jakosti užití se týkají dopadu tohoto systému na zúčastněné strany.

Type of properties measured	Software product properties	Computer system behaviour properties	Human-computer system impact properties
Type of quality measure	Internal: inspection of static properties	External: test or modelling of dynamic properties	Quality in use: test or observation of results of real or simulated use
Type of properties of software product	Inherent	Computer system-dependent	Human-computer system-dependent
Type of properties of computer system		Inherent	Human-computer system-dependent
Type of properties of human-computer system			Inherent

Tabulka 1: Rozdíly mezi mírami interní jakosti, externí jakosti, a jakosti užití. Zdroj: [ISO/IEC 2009a].

Vnitřní míry software mohou být použity v rané fázi procesu vývoje systému/software pro predikci externích měr jakosti systému/software. Existují často vnitřní a vnější míry pro stejné vlastnosti.

4.2 Požadavky na jakost software

Požadavky na jakost software musí vycházet z reálných potřeb zúčastněných stran. Funkce měření představuje dle Vaníčka [Vaníček 2006] interpretaci vlastnosti kvality software a cílová hodnota měření kvality představuje požadavky na kvalitu. Podobně, skutečná hodnota měření kvality představuje pozorovanou kvalitu software.

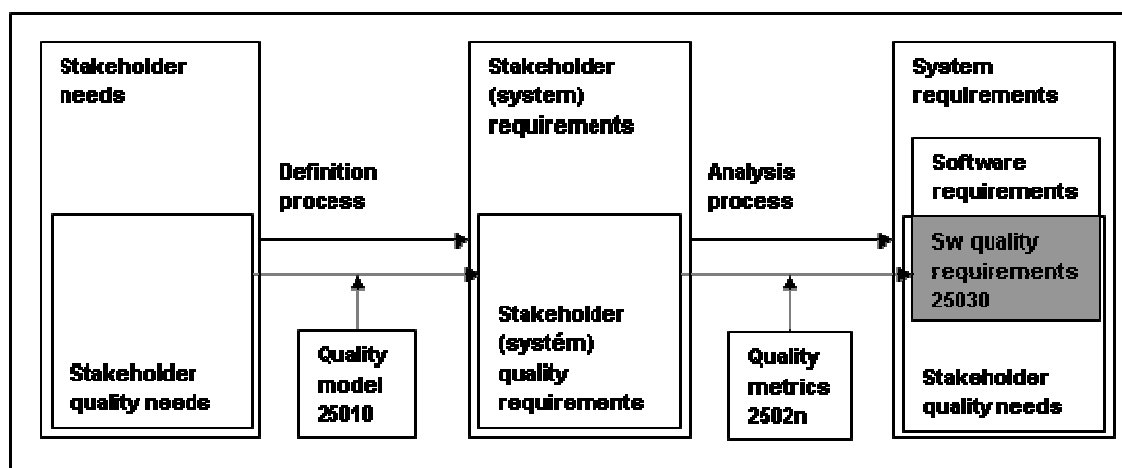
Požadavky na kvalitu softwaru, stejně jako všechny ostatní požadavky nelze chápat izolovaně, ale pohled musí být v širších souvislostech. Požadavky na kvalitu softwaru mají zvláštní úzký vztah k funkčním požadavkům. Funkční požadavky hrají důležitou roli pro určení požadavků na kvalitu software. Funkčnost je jednou z charakteristik pro vnitřní i vnější jakost v ISO/IEC 9126-1 a ISO/IEC 25010. Požadavky na funkčnost by neměly být zaměňovány s funkčními požadavky. Funkčnost je schopnost software poskytovat funkce, které splňují funkční požadavky. Požadavky na funkčnost jsou zapracované do požadavků na software, který má být vhodný, přesný, interoperabilní a zabezpečený v souladu s příslušnými funkčními normami a předpisy [ISO/IEC 2006a].

V některých situacích [ISO/IEC 2006a] je smysluplné zadání požadavku na jakost softwarového produktu, zatímco v jiných situacích se jakost vztahuje pouze na část softwarového produktu. Například, některé funkce jsou relevantní pouze pro konkrétní uživatele a mají specifické požadavky na kvalitu, které jsou odlišné

od požadavků na jakost pro další funkce určené pro jiné účely a jiné uživatele. Je tedy důležité určit, které části softwarového produktu jsou relevantní pro požadavky na jakost softwaru. Jinými slovy, požadavek na kvalitu souvisí s částí softwarového produktu. Například, některé funkce jsou určeny pro koncové uživatele, a proto je nutné požadovat nízkou toleranci chyb, zatímco další skupina funkcí může být určena pro odborníky, a tím lze umožnit větší toleranci chyb. V obou případech by se měl mechanismus chybové tolerance a stupeň požadované tolerance chyb řádně stanovit.

Jakost užití je definována do jaké míry výrobek splňuje potřeby stanovené uživateli k dosažení stanovených cílů s účinností, produktivitou, bezpečností a spokojeností v určitém kontextu použití. Proto jsou požadavky na jakost užití úzce spojeny s jinými požadavky na systém, jako jsou požadavky na hardware, obchodní požadavky a požadavky na koncové uživatele [ISO/IEC 2006a].

Funkce měření umožňuje interpretaci vlastností kvality softwaru, tj. přiřazuje hodnotu vlastnosti. Cílová hodnota tohoto měření kvality softwaru představuje požadavky na jakost software, tedy to, co je požadovanou hodnotou vlastnosti. Podobně, skutečná hodnota měření kvality představuje pozorované kvality software [ISO/IEC 2006a].



Obrázek 16: Vymezení a analýza požadavků na kvalitu software. Zdroj: [ISO/IEC 2006a].

Obrázek 16 ukazuje, jak jsou požadavky na kvalitu software odvozené jako součást požadavků na procesy, které jsou definovány v ISO/IEC 15288. Definice procesu se zaměřuje na požadavky zúčastněných stran na systém. Proces analýzy předpokládá, některá architektonická rozhodnutí, která umožňují identifikovat požadavky relevantní pro prvky software systému. Při zaměření na kvalitativní potřeby zúčastněných stran na kvalitu modelu, které jsou definovány v ISO/IEC 25010, jsou užitečné pro vymezení

požadavků na kvalitu zúčastněných stran. Podobně metriky definované v ISO/IEC 2502n jsou vhodné pro formalizování požadavků zainteresovaných stran na kvalitu software [ISO/IEC 2006a].

4.2.1 Kategorizace požadavků na systém

Systém se skládá z několika vzájemně se ovlivňujících prvků. Mohou být definovány a kategorizovány různými způsoby. Na Obrázku 17 je znázorněno podobné členění systémových požadavků. Systémové požadavky mohou zahrnovat například požadavky na software, počítačový hardware, data, mechanický systém, business organizace, apod., a mohou pocházet od různých zúčastněných stran, včetně uživatelů, organizace a úředních subjektů. Požadavky na software se zabývají buď softwarovým produktem, nebo procesem vývoje softwaru. Požadavky na softwarový produkt zahrnují funkční požadavky, požadavky na kvalitu a manažerské požadavky. Funkční požadavky zahrnují aplikační doménu specifických i funkčních požadavků, které podporují požadavky na kvalitu. Požadavky na kvalitu mohou také znamenat, architektonické a strukturální požadavky [ISO/IEC 2006a].

Požadavky na vývojový proces software mohou například zahrnovat požadavky na artefakty, proces, projekt, rozvoj organizace a vývojáře. Tam budou často závislosti mezi požadavky na vývoj software a požadavky na softwarový produkt [ISO/IEC 2006a].

System requirements	Software requirements	Software product requirements	Inherent property requirements	Functional requirements		
			Assigned property requirements	Quality requirements	Quality in use	
		Software development requirements		Development process requirements	External quality	
			Development organisation requirements			
	Other system requirements	Include for example requirements for hardware, businesses, business processes, and end users				

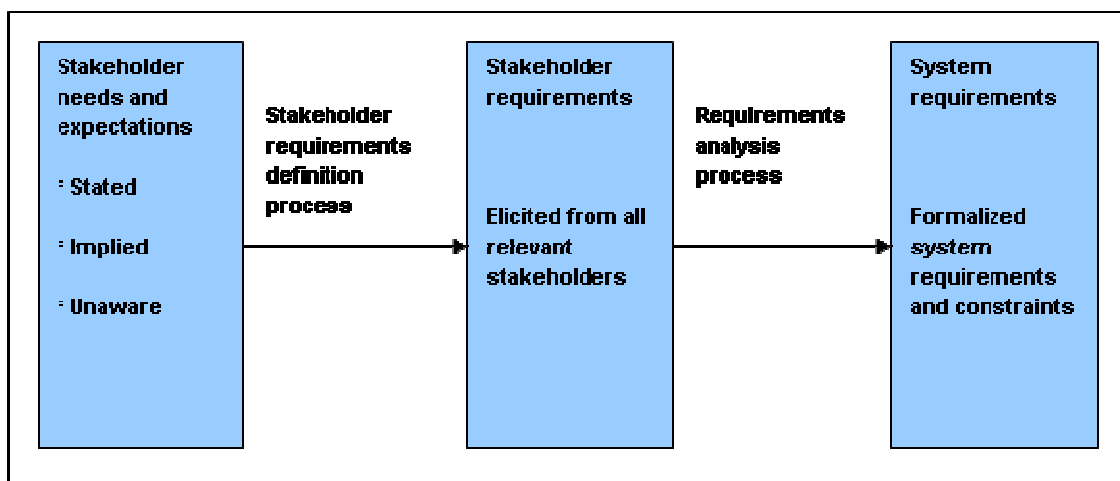
Obrázek 17: Kategorizace požadavků na systém. [ISO/IEC 2006a].

4.2.2 Zúčastněné strany a požadavky zúčastněných stran

Systémy mají různé zúčastněné strany [*stakeholders*], které mají o systém zájem po celou dobu jeho životního cyklu. Zúčastněné strany systému zahrnují všechny osoby (například koncové uživatele), organizace (např. organizace koncového uživatele) a orgány (např. zákonné a regulační orgány), které mají oprávněný zájem o systém. Zúčastněné strany mají různé potřeby a očekávání od systému. Jejich potřeby a očekávání se mohou během životního cyklu systému měnit [ISO/IEC 2006a].

Potřeby zainteresovaných stran mohou být explicitně uvedeny, nebo pouze předpokládány. Předpokládané potřeby jsou často odvozené z kontextu, kdy softwarový produkt má být použit a reprezentovat očekávání na základě podobných softwarových produktů nebo stávajících pracovních postupů, běžných pracovních postupů a činnosti podnikání, zákonů a předpisů, atd. Zainteresovaným stranám nejsou někdy známy všechny jejich potřeby. V mnoha situacích se tak potřeby zainteresovaných stran projeví, až když softwarové produkty a související obchodní procesy nebo úkoly mohou být vyzkoušeny. Scénáře, use cases, a prototypy jsou příklady metod, které mohou být použity k identifikaci předpokládaných a nevědomých potřeb v raném stadiu vývoje projektu. V některých případech se skutečné potřeby některých zúčastněných stran liší od toho, co vyjadřují [ISO/IEC 2006a].

Potřeby zainteresovaných stran a očekávání jsou identifikovány prostřednictvím elicitací požadavků a definice procesů, jak je znázorněno na obrázku 18. Tento proces musí vzít v úvahu všechny potřeby, očekávání a touhy zúčastněných stran. Toto zahrnuje potřeby a požadavky uložené společnostmi, omezení způsobená nabyvatelem, a potřeby koncových uživatelů. V některých situacích mají zúčastněné strany protichůdné potřeby. Konflikty mezi požadavky zúčastněných stran by mohly například být mezi různými perspektivami koncového uživatele, nebo mezi potřebami nabyvatele a dostupnými znalostmi, zkušenostmi či zdroji v developerských organizacích. Výsledkem definice procesu jsou požadavky zúčastněných stran, výsledkem analýzy jsou požadavky na systém [ISO/IEC 2006a].



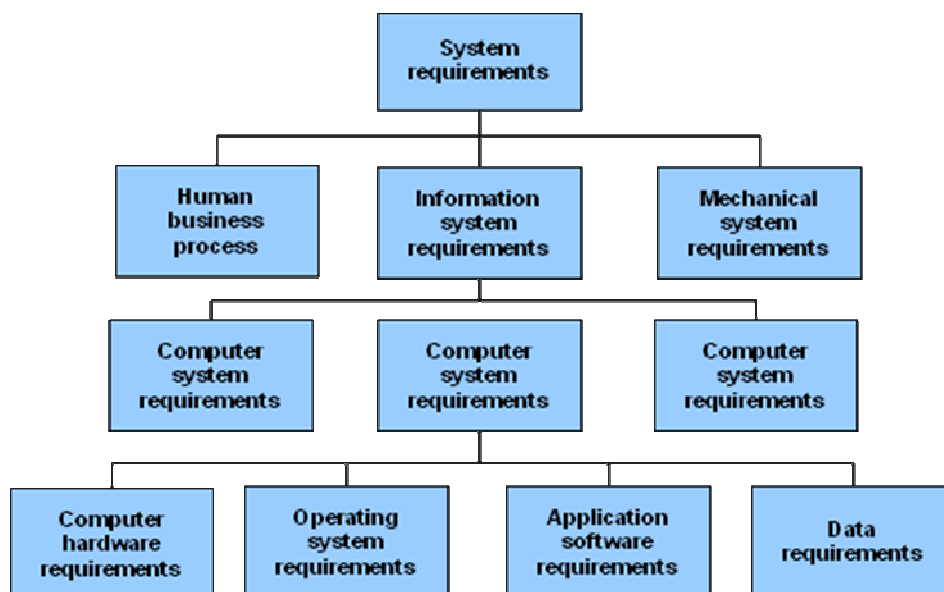
Obrázek 18: Definice a analýza požadavků zúčastněných stran. [ISO/IEC 2006a].

4.2.3 Požadavky zúčastněných stran a požadavky na systém

Analýza procesu transformuje požadavky zainteresovaných stran do technického pohledu požadavků na systém, které mohou být použity k realizaci požadovaného systému, viz obr. 18. Technický pohled na potřeby je označován jako systémové požadavky. Systémové požadavky jsou ověřitelné a uvádějí, které vlastnosti systém má mít, aby byly splněny požadavky zúčastněných stran [ISO/IEC 2006a].

Systém se často skládá z různých prvků, z nichž každý má specifické vlastnosti a slouží k různým účelům v celém systému. Má-li být funkční, systémové požadavky musí být formulovány jako požadavky na různé prvky systému. Jak různé prvky ovlivňují nabízené možnosti systému, požadavky pro různé prvky systému nelze posuzovat izolovaně, ale pouze v širším pohledu, včetně požadavků na ostatní prvky systému [ISO/IEC 2006a].

Požadavky zúčastněných stran mohou být realizovány alternativními způsoby, například, buď v hardware nebo v software, nebo jako podnikové business procesy. Taková implementační rozhodnutí jsou součástí procesu návrhu na vysoké úrovni. Obrázek 19 ukazuje hierarchii požadavků na základě rozhodnutí návrhu systému [ISO/IEC 2006a].



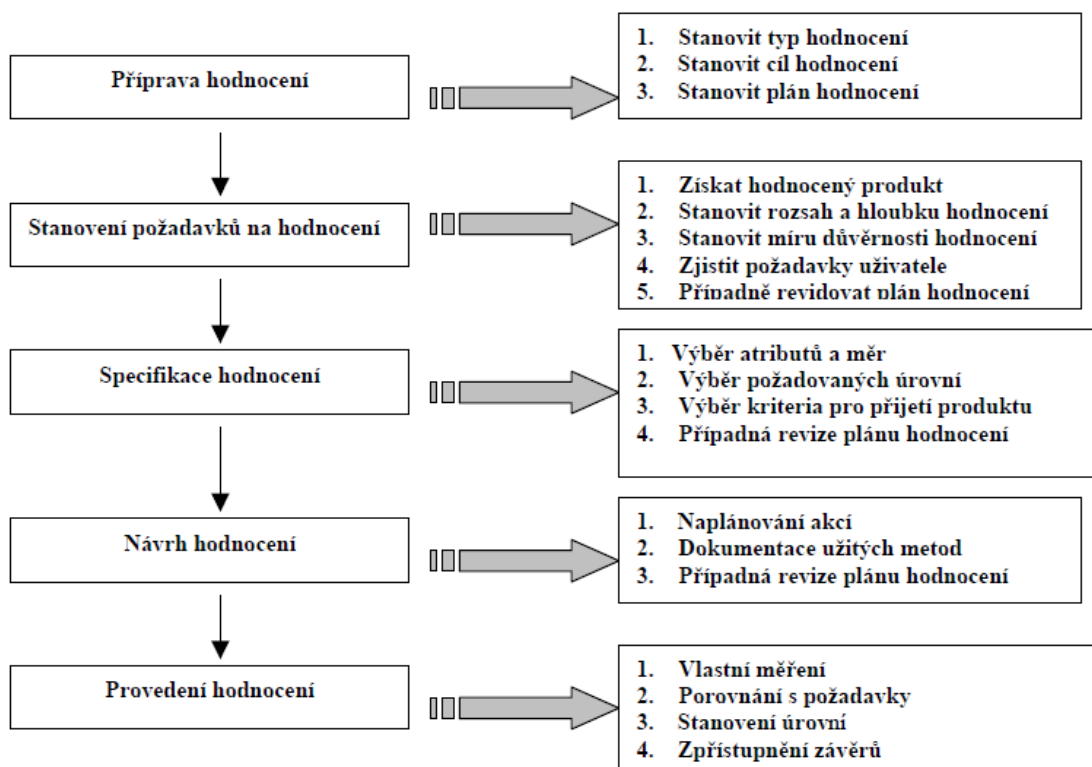
Obrázek 19: Hierarchie - systém a požadavky na software. Zdroj [ISO/IEC 2006a].

4.3 Postup hodnocení jakosti

Hodnocení je třeba [Vaníček 2004a] provést systematicky, podle předem popsaneho a schvaleneho plánu. Sestává z několika kroků, z nichž žádný nelze vynechat. V případě, že ten, kdo hodnocení vyžaduje, není totožný s tím, kdo je provádí, je třeba výsledky každé z následujících etap odsouhlasit mezi objednavatelem a realizátorem hodnocení. Kroky hodnocení jakosti jsou:

- Stanovení požadavků na jakost.
- Specifikace a plán hodnocení jakosti.
- Vytvoření a odsouhlasení plánu hodnocení.
- Vlastní hodnocení.
- Posouzení výsledků.

Pro hodnocení je třeba dodržet jeho životní cyklus znázorněný na následujícím obrázku číslo 20. Stanoví i náležitosti hodnocení, které jsou pro všechny uvedené pohledy společné. Hodnocení může sloužit jak ke zvýšení a Zabezpečení kvality při vývoji produktu, k rozhodnutí, zda při vývoji postoupit k další etapě, přijmout či odmítnout dílčí produkt, tak i k rozhodnutí opatřovatele o uzavření kontraktu nebo k výběru mezi několika alternativami.



Obrázek 20: Referenční model hodnocení kvality softwarového. Zdroj [Vaníček 2008c].

4.4 Míry atributů

V současné době je [Vaníček 2004a] jen z oblasti mezinárodní normalizace navrženo několik set atributů a měř, které více či méně popisují aspekty důležité pro jakost. Názory, zda tyto míry popisují něco rozumného či pro jakost podstatného, se mohou legitimně lišit případ od případu. Jde o následující skupiny problémů:

- jak na základě modelu jakosti zjistit vhodné kandidáty na měřitelné atributy jakosti,
- jak na základě zvolených atributů stanovit data, která budeme muset sbírat či zjišťovat, a zdroje těchto dat,
- jak hodnotit možnost užití zjištěných přímých měř pro stanovení odvozených měř a měř zjištěných pro komponenty ke stanovení měř celku,
- na jaké otázky hodnoty zjištěných měř odpovídají a v kterých etapách životního cyklu projektu a kterými profesemi je možno získaných měř využít.

V souladu s normou ISO/IEC 15939 se rozlišují **základní míry** [*base measures*] a **odvozené míry** [*derived measures*]. Základní míry stanovíme přímo z reality, přímým pozorováním nebo vyhodnocením experimentu. Odvozené pak výpočtem (vzorcem či

algoritmem) ze základních, ale i jiných odvozených měř. Ukazuje se, že **základní** (přímé) **míry** v oblasti informačních systémů a softwarových produktů mohou být pouze trojího typu:

- **Počet** (celé nezáporné číslo) prvků nějaké množiny.
- **Doba** (reálné číslo) trvání nějakého jevu.
- **Kategorie** (zpravidla vyjadřovaná též celým číslem), do které je jev nebo entita zařazena. Kategorie mohou být uspořádány podle preferencí některým typem uspořádání. Nejsou-li uspořádány, znamená hodnota míry pouze označení kategorie, do které je entita zařazena a nevypovídá nic o našich preferencích.

Jiné typy základních měř se u informačních systémů a softwaru v souvislosti s jakostí nevyskytují. Počet vede na přirozené měření v měřicí stupnici poměrového typu. Jednotkou může být buď jeden prvek nebo stanovené množství prvků (například kilobytů nebo počet řádků na stránce dokumentace). Také doba vede na stupnici poměrového typu. Kategorie pak na ordinální nebo nominální typ měřicí stupnice.

Odvozené (nepřímé) **míry** se vytvářejí ze základních užitím nějakých funkcí a matematických operací. Odvozování může probíhat v řadě úrovní. Z odvozených měř lze provedením dalších operací získat další odvozené míry. Typickým a častým způsobem získání odvozených měř může být například:

- Porovnání míry s nějakou etalonní hodnotou (přirozeným maximem – ideální hodnotou nebo očekávanou hodnotou, která „ještě vyhovuje“), měřenou v téže měřicí stupnici. Pokud je odvozená míra definována jako poměr nebo procenta (to bývá častější), dostaneme míru v absolutní stupnici. Tato míra má však smysl pouze, když vstupní míry jsou ve stupnicích poměrového nebo absolutního typu. Jde-li pouze o míry ordinálního nebo nominálního typu, smysl nemá. Pokud stanovíme rozdíl (což bývá méně časté), dostáváme míru ve stupnici diferenční nebo intervalové (opět za příslušných omezení na typy stupnic vstupních měř).
- Zjišťování počtu nějakých jevů za danou časovou jednotku. To vede k odvozené míře ve stupnici poměrového typu.

V každém případě je třeba nepodcenit otázku, zda má odvozená míra definovaná nějakým vzorcem empirický význam, a v jakém typu měřicí stupnice provádí měření. Pro tyto míry, které jsou pro měření jakosti potřebné, avšak samy o sobě o jakosti přímo nic nevypovídají, se v rámci projektu SQuARE zavádí pojem **prvky pro měření QME**

[*Quality Measure Elements*]. Pro užitečnost modelu je důležité, aby prvků pro měření nebylo příliš mnoho. Jejich sledování totiž s sebou nese náklady, které péči o kvalitu prodraží. Rozumně rozsáhlá množina prvků pro měření, která by však pokryla zdroje dat pro výpočet měř navrhovaných atributů však v rámci SQuaRE nebyla dosud specifikována.

4.4.1 Výběr atributů a měř

Při výběru atributů a měř bychom měli brát v úvahu především tyto aspekty [Vaníček 2004a]:

- relevance atributů a měř k informační potřebě pro hodnocení jakosti,
- zda aktuální stav životního cyklu produktu koresponduje se stavem, kdy jsou požadované údaje k dispozici,
- náklady spojené se sběrem potřebných dat a dostupnost lidských zdrojů pro tuto práci,
- existence podpůrných nástrojů pro sběr potřebných dat a jejich zpracování,
- ochota příslušného personálu poskytnout data a provozní problémy, které může sběr potřebných dat vyvolat,
- narušení „soulkromí“ spojená s měřením,
- zkušenosti, které dosud máme s danou mírou a jejím vztahem k jakosti,
- zda jsou požadovaná data použitelná obecně, i pro jiné míry, které budeme potřebovat pro měření dalších charakteristik jakosti produktu,
- zda je daný atribut „kontextově závislý“, to znamená, do jaké míry jeho hodnota závisí i na jiných faktorech, než jsou faktory jakosti, které nás zajímají,
- zda je možné zajistit opakovatelnost měření se stejnými nebo podobnými výsledky,
- zda je míra „odolná“ vůči účelovému ovlivnění stranou, která má na výsledku měření zájem. Je znám případ, kdy v okamžiku, v kterém bylo rozhodnuto hodnotit výkon programátorů počtem napsaných a odzkoušených příkazů zdrojového kódu a tato zásada vyhlášena, zaznamenala téměř okamžitě délka programů bouřlivý nárůst.

V případě, že měření hodnocení provádí jiný subjekt než je ten, který má na hodnocení jakosti zájem, je třeba prověřit výsledky obou předchozích kroků společnou revizí obou zúčastněných stran. Při měření je pak třeba držet se předem stanoveného plánu a jednotlivé kroky dokumentovat. Vytvořené datové soubory je třeba verifikovat, zda

odpovídají skutečnosti, a analyzovat je, pokud jde o jejich úplnost a bezrozpornost. Důležitá je i nestrannost při prezentaci výsledků hodnocení jakosti toho, kdo na základě hodnocení jakosti bude rozhodovat.

4.4.2 Vlastnosti atributů a měř

Atributy jakosti a jejich míry jsou v normalizačních technických zprávách striktně děleny [Vaníček 2004a] podle jednotlivých charakteristik a podcharakteristik jakosti, takže každá z nich je zařazena pod jedinou charakteristiku a podcharakteristiku.

Protože některé atributy mohou být považovány jak za vnější, tak za vnitřní, pouze v závislosti na tom, jaké datové vstupy se používají pro jejich výpočet, je vhodné skutečnost, zda jde o vnější či vnitřní atribut jakosti nebo o atribut jakosti používání, považovat rovněž za vlastnost atributu a příslušné míry.

Nyní budou uvedeny jednotlivé obecné vlastnosti, které je doporučeno u měř sledovat taxativně:

1. Jméno měřeného atributu,

doporučuje se uvádět několik slov, která příslušný atribut charakterizují dostatečně dobře a jednoznačně, spolu s případnou zkratkou.

2. Jméno užití míry,

pokud je pro daný atribut navržena jediná míra, může být jméno míry s jménem atributu totožné. Často je však vhodné měřit též atribut různými mírami, případně i mírami v různých typech měřicí stupnice. V tomto případě je vhodné míry rozlišit, nejlépe doplněním vhodných slov na závěr (například: funkční pokrytí *základní*, funkční pokrytí *normované*). Pokud s mírou dále pracujeme ve vzorcích (to je třeba například pro stanovení klasifikace charakteristik či podcharakteristik jakosti), je vhodné stanovit pro míru i vhodnou zkratku (identifikátor).

3. Účel míry,

ten je nejlépe popsat otázkou, na kterou daný atribut či zjištěná míra odpovídá.

4. Metoda měření,

zde se stručně popíše postup, který vede ke stanovení hodnoty míry atributu.

5. Datové prvky a vzorec pro výpočet míry,

zde se popíše datové prvky, ze kterých se míra určuje, a vzorec výpočtu hodnoty této míry z těchto datových prvků.

6. Interpretace hodnot míry,

zde se popíše množina hodnot, kterých mohou míry nabývat (při měření čísla to obvykle bývá číselný interval) a uvede se výrok o tom, kdy indikuje míra vyšší jakost. Příkladem může být následující údaj: „ $0 \leq X \leq 1$, čím blíže k 1, tím lépe“.

7. Typ měřicí stupnice,

při měření čísla se obvykle uvádí příslušný typ podle Stevnsy klasifikace. Při zobecněném měření nebo při netradičním typu měřicí stupnice lze uvést množinu přípustných transformací hodnot měř.

8. Typ měřených hodnot,

u základních měř nejčastěji *počet* (poruch, chyb, souborů, dokumentů, řádek zdrojového kódu, proměnných, funkcí, ...) nebo *čas* (výpočtu, bezporuchového provozu, ...). U odvozených například *počet / počet*, *počet / čas* nebo *čas / čas*.

9. Zdroj dat pro určení míry,

zde se uvede, co bude zdrojem dat pro měření. Může to být například specifikace systému, uživatelská dokumentace k systému, zdrojový kód programu, protokol o testování systému, evidence rutinního využívání systému apod.

10. Etapa životního cyklu, v které lze měření provádět,

zde se uvede nejranější etapa životního cyklu, v které již jsou k dispozici data, potřebná pro vyčíslení míry. Tedy například specifikace systému, návrh systému, implementace softwaru, odzkoušení systému, provoz systému, údržba systému. Doporučuje se užívat modelu životního cyklu popsaného v normě **ČSN ISO/IEC 12207 „Informační technologie – Model životního cyklu softwaru“** [*Information technology – Software life cycle model*] z roku 1995, případně i s přesným odkazem na příslušný odstavec této normy.

11. Profese, využívající výsledky měření,

zde se uvede, které profesní skupiny pracovníků bude zjištěná míra převážně zajímat, a které mohou výsledky ve své práci využít. Typické zde uváděné skupiny pracovníků jsou: opatrovatel produktu, uživatel, provozovatel, dodavatel, projektant, manažer jakosti apod.

12. Charakteristiky a podcharakteristiky jakosti podstatně ovlivněné danou mírou,

lze uvést i několik charakteristik a podcharakteristik.

13. Charakteristiky a podcharakteristiky jakosti částečně ovlivněné danou mírou,

lze uvést i několik charakteristik a podcharakteristik.

14. Druh míry - vnější, vnitřní nebo pro jakost užití.

4.5 Měřicí stupnice

Zde budou uvedeny nejznámější typy v případě, že formální strukturu pro měření tvoří číselné obory \mathfrak{R} nebo \mathfrak{R}^+ , tedy po řadě obory všech reálných čísel nebo všech kladných reálných čísel. Základní typy jsou známy též jako Steversona klasifikace typů měřicích stupnic a jsou uvedeny v následujících podkapitolách [Vaníček 2004a].

Klasická Steversona stupnice zřejmě nevystihuje všechny situace, které mohou při měření pomocí čísel nastat. Pro účely tohoto dokumentu však postačuje a další typy číselných měřicích stupnic nebudou dále popsány.

4.5.1 Absolutní stupnice

Formální struktura a přípustné transformace: formální strukturou může být buď množina \mathfrak{R} (všechna reálná čísla) nebo množina \mathfrak{R}^+ (kladná reálná čísla), s přirozeným uspořádáním podle velikosti a operací sčítání čísel, množinu přípustných transformací tvoří pouze identické zobrazení. Všechny ostatní transformace přípustné nejsou. Při měření není žádná volnost.

Nejčastěji se používá tam, kde se snažíme vyjádřit měřenou hodnotu poměrem k nějakému etalonu nebo přirozenému maximu. Například poměr počtu (nebo procento) realizovaných funkcí k počtu všech požadovaných funkcí informačního systému, poměr nebo procento komentovaných příkazů ve zdrojovém kódu programu k počtu všech příkazů kódu.

Výhodou je snadná přehlednost pro porovnávání a nezávislost na jednotkách, v kterých se měření provádí. Hodnota míry je vždy bezrozměrné číslo. Nevýhodou je, když při skládání celku z komponent nelze přímo, bez dodatečných informací, vypočítat míru celku pouze na základě měř jeho komponent. Při porovnávání počtu hrozí, že velmi významné skutečnosti ovlivní míru stejnou měrou jako nevýznamné.

4.5.2 Poměrová stupnice

Formální struktura a přípustné transformace: formální strukturou může být množina \mathfrak{R}^+ (kladná reálná čísla), s přirozeným uspořádáním podle velikosti a operací sčítání čísel. Množinu přípustných transformací tvoří zobrazení tvaru $f(x) = a * x$, kde $a > 0$.

Nejčastěji užívaná míra pro struktury, kde je definováno slučování a toto slučování má vlastnosti, které ji dovolují zavést. Užití je rozsáhlé, například: délka

a hmotnost předmětu, průměrná doba bezporuchové práce informačního systému, doba potřebná pro vyškolení obsluhy, náklady na vývoj, počet řádků zdrojového textu kódu programu, atd. Jde vždy o měření, u kterých jediná volnost, kterou máme, je volba základní jednotky, s kterou se provádí porovnání (metry, kilometry, yardy či versty, gramy, kilogramy). Hodnota má fyzikální rozměr jednotky, s kterou se měřený objekt porovnává.

Jde o velmi výhodnou stupnici, umožňující snadný výpočet míry celku ze známých hodnot měř komponent. Prakticky všechna „rozumná“ zjištění ve formální struktuře (pro čísla), lze přenést do struktury empirické. U jakosti informačních systémů a měř pro softwarové inženýrství bývá cílem tohoto typu stupnice dosáhnout. Nevýhodou je, že ne vždy je možné měření v měřicí stupnici tohoto typu nalézt.

4.5.3 Intervalová stupnice

Formální struktura a přípustné transformace: formální strukturou je množina \mathfrak{R} (všechna reálná čísla) s přirozeným uspořádáním podle velikosti a operací sčítání čísel, která se však nezachovává při měření plně. Tento typ stupnice nelze realizovat, je-li nosičem formální struktury množina všech kladných čísel \mathfrak{R}^+ . Množinu přípustných transformací tvoří zobrazení tvaru: $f(x) = a \cdot x + b$, kde $a > 0$ a b jsou reálná čísla.

Typickým příkladem takové stupnice je stupnice teploty. Zde je proměnlivá jak počáteční „nula“, tak měrná jednotka, s kterou je porovnávána změna teploty. Jiný příklad je měření aktuálního času, kde může být proměnlivá jak jednotka doby, tak počátek měření, nebo měření odchylky aktuální hodnoty od očekávané. Pro hodnocení jakosti informačních systémů a softwarových produktů se užívají spíše výjimečně.

Výhody a nevýhody jsou obdobné jako u poměrové stupnice (viz předchozí kapitola). Empirický význam má z běžně užívaných statistických momentů pouze aritmetický průměr a vše závislé jen na pořadí, například medián. V řadě případů však nelze realizovat.

4.5.4 Ordinální stupnice

Formální struktura a přípustné transformace: pro tento typ měřicí stupnice není třeba v empirické struktuře uvažovat operaci skládání. Formální strukturou může být buď množina \mathfrak{R} (všechna reálná čísla) nebo množina \mathfrak{R}^+ (kladná reálná čísla), s přirozeným uspořádáním podle velikosti, ale také jen množina všech celých nebo všech přirozených

čísel nebo podmnožiny těchto množin. Například pouze celá čísla nebo celá čísla z nějakého intervalu. Sčítání čísel ve formální struktuře není třeba uvažovat. Výpočty, které jej užívají, však nevedou k závěrům, které by bylo přípustné interpretovat v empirické struktuře. Přípustné transformace jsou dány množinou všech (ostře) rostoucích funkcí f , to je funkcemi, pro které platí pro všechna x a y : $x < y \Rightarrow f(x) < f(y)$. Zaměníme-li jednu (a jen jednu) z relací v empirické či formální struktuře za opačnou, změní se ovšem požadavek na funkci f na to, aby byla (ostře) klesající.

Lze použít u všech hodnocení, která se opírají o subjektivní odhad a pouhé porovnání bez zřetelné kvantifikace. Patří sem například klasifikace výkonu studentů u zkoušky, nebo různá hodnocení typu „naprosto souhlasím, souhlasím, spíše souhlasím, jsem na rozpacích, spíše nesouhlasím, nesouhlasím, naprosto nesouhlasím“, s kterými se setkáváme v různých anketách. Takových situací je při hodnocení jakosti nesporně převážná většina.

Nespornou výhodou je, že tuto stupnici většinou není problém sestrojít a použít téměř ve všech situacích souvisejících s informačními systémy a softwarovými produkty. Nevýhodou je značně omezená možnost interpretace výsledků získaných výpočtem. Nemožnost vypočítat míru celku na základě měř komponent, ze kterých se celek skládá; v obecném případě dokonce i nemožnost usoudit, že zvýšením míry některé komponenty nedojde ke snížení míry celku. Pro možnost takovéhoho úsudku, který je pro řízení jakosti v průběhu vývoje nezbytný, je třeba, aby měření splňovalo další předpoklady. Konkrétně jde o monotónnost míry vzhledem k operaci slučování v empirické struktuře. Ani tuto vlastnost ordinální typ měřicí stupnice obecně negarantuje.

4.5.5 Nominální stupnice

Formální struktura a přípustné transformace: u tohoto typu stupnice neuvažujeme v empirické ani ve formální struktuře žádné relace ani operace. Nelze však ani užít závěry získané ve formální struktuře výpočtem či porovnáváním čísel podle velikosti. Místo čísel bychom mohli se stejným výsledkem užít třeba písmen nebo prvků jakékoliv množiny. Přípustné transformace jsou dány množinou všech prostých funkcí f , to je funkcí, pro které platí: $x \neq y \Rightarrow f(x) \neq f(y)$.

Použití: pouze pro klasifikaci do tříd, bez vytváření jakýchkoli závěrů o prioritách mezi těmito třídami. Například druhy chyb v programu lze dělit

na syntaktické a sémantické, syntaktické dále, atd. Nehodnotí se při tom, která z chyb je závažnější, či více zavrženíhodná.

Výhody: Žádné předpoklady pro existenci takového měření. Nevýhodou je nemožnost interpretace netriviálních závěrů z formální struktury do empirické.

5 Audit informačního systému

Pokud předchozí část práce byla věnována jakosti informačních systémů a jejímu hodnocení, nelze vynechat z důvodu komplexnosti přehledu téma auditu informačních systémů. Audit informačního systému lze chápat jako nástroj umožňující preventivní působení na řízení informačních technologií a snižovat tak rizika z jejich využívání, a monitorování procesů probíhajících v těchto systémech.

V této kapitole bude popsán současný stav způsobů provádění auditu informačních systémů, který vychází z publikace [Svatá 2011]. Text je zaměřen zejména na obecný postup provádění auditu IS/IT.

Audit informačního systému je specifický proces, který se zabývá posuzováním a poradenstvím objektů v prostředí, kde se používají informační technologie. Jeho cílem je kvalitativně a/nebo kvantitativně přispět ke správné organizaci informačního systému tak, aby byly splněny požadavky uživatelů, zákonů, smluv či jiných regulací (externích či interních). Objekty mohou být organizace a řízení IS, základní i aplikační software, technické vybavení, telekomunikační systémy, procesy tvorby a údržby systémů, ochrana a bezpečnost systému, data (databáze) apod. [Svatá 2011].

Důležitou součástí auditorské činnosti představují standardy, protože s jejich pomocí lze splnit jeden ze základních požadavků auditu, kterým je jeho objektivita. Standardy obecně fungují jako kritéria hodnocení, proti kterým auditor porovnává skutečnost, a která současně objektivizují jeho hodnocení. Mezi organizace, které se zabývají auditem IS ve větší míře, patří především ISACA (*Information System Audit and Control Association*) a INTOSAI (*International Organisation of Supreme Audit Institutions*). ISACA vydává standardy, které jsou závazné pro interní a externí auditory a dělí se na: standardy [*Standards*], návody [*Guidelines*] a procedury [*Procedures*]. INTOSAI zastřešuje a sdružuje organizace provádějící externí audity vládních organizací. Pro svoje standardy vytvořila nový rámec, který je určující pro všechny organizace, které se zabývají primárně finančním auditem [Svatá 2011].

5.1 Některé metodiky auditu IS/IT

Každý audit IS má ve své podstatě charakter projektu, a tedy se opírá o určitý životní cyklus, který je obvykle součástí metodik. Těch existuje celá řada, v této kapitole budou vybrané metodiky nastíněny. Jejich detailní analýza však není předmětem této práce.

Nezastupitelné postavení má dle [Svatá 2011] v obecných mezinárodních metodikách pro audit IS metodika COBIT a především její jedna část – dokument IT Assurance Guide (Návod pro ujištění v oblasti IT). Pro auditory COBIT poskytuje dva druhy návodů: IT Governance Implementation Guide: Using Cobit and Val IT, který je založen na kontrolních praktikách, pomáhajících zavést kontrolní cíle do praxe, a IT Assurance Guide: Using Cobit, který je založen na etapách a činnostech, které jsou společné každému auditu, pokud jej chápeme jako projekt s určitým životním cyklem. COBIT (*Control Objectives for Information and related Technology*) byl dle [COBIT 2007] vyvinut jako všeobecně přijímaný standard pro správné postupy řízení, kontroly a auditu informačních technologií. Řízení a správa podniku, jako systém, kterým je organizace vedena a kontrolována (*Enterprise Governance*) a řízení a správa podnikové informatiky, jako systém, kterým je IT v organizaci vedeno a kontrolováno (*IT Governance*), jsou podle COBIT vzájemně podmíněné systémy.

Základní princip metodiky COBIT je postaven na propojení tří různých aspektů řízení informačních technologií v organizacích. Dle [Svatá 2011] jsou jimi:

- cíle organizací ve formě požadavků na vlastnosti (kritéria) dodávaných informací; uvažují se následující informační kritéria: efektivnost, výkonnost, důvěryhodnost, integrita, dostupnost, shoda, hodnověrnost,
- zdroje informačních technologií; jako základní zdroje se uvažují aplikace, informace, infrastruktura a lidé,
- procesy; mapa procesů vychází ze čtyř domén, které kopírují životní cyklus řízení IT a jsou označovány zkratkou : PO (*Plan and Organice*), AI (*Acquire and Implement*), DS (*Deliver and Support*) a MI (*Monitor and Evaluate*).

Dokument IT Assurance Guide je primárně určen auditorům IS a „assurance“ profesionálům. Jeho cílem je poskytnout návody pro jednotlivé etapy a činnosti životního cyklu auditu a ujištění. Obsahově navazuje na Cobit 4.1 a umožňuje tak provést hodnocení (audit/ujištění) procesu vzhledem k těmto doporučeným cílům. Vlastní postup projektu ujištění dokument IT Assurance Guide popisuje pomocí životního cyklu, který se skládá ze tří základních etap: plánování, stanovení rozsahu a realizace. Každá z těchto etap je dále rozpracována do dílčích činností a celý model se označuje termínem IT Assurance Map [Svatá 2011].

Dokument Val IT (*Value IT*) je odpovědí na poptávku praxe, která požadovala praktický nástroj pro hodnocení, identifikaci a alokaci investic v oblasti IS/IT tak, aby byla v souladu se zásadami IT Governance (aby dokázala hodnotit přidanou hodnotu pro business plynoucí z implementace IT). Dokument je úzce spojen s metodikou Cobit. Val IT doporučuje osvědčené postupy přispívající k hodnocení cílů procesů – přidané hodnoty z IT. Doplnuje tedy Cobit z obchodního a finančního hlediska. Klíčovým pojmem dokumentu je hodnota IT investice. Za tuto hodnotu se nepovažuje pouze přínos investice, který nalezneme na konci jejího cyklu, ale hodnotou se rozumí přínos investice po celou dobu jejího životního cyklu. Současně se nejedná o hodnotu finanční, ale v úvahu se berou zejména nefinanční přínosy protože právě ty jsou pro investice v IT důležité. Metodika pracuje se třemi hlavními termíny: projekt, program a portfolio. Projekt definuje jako strukturovanou sadu aktivit týkajících se dosažení definované úrovně, která je nezbytná pro dosažení určitého business cíle (výstupu), a která se řídí odsouhlaseným plánem a rozpočtem. Program je struktura vzájemně provázaných projektů, které jsou jak nezbytné, tak i dostačující pro dosažení požadovaného výstupu v podobě vytvořené hodnoty. Pro Val IT je základní jednotkou řízení investic investiční program. Portfolio je zde chápáno jako seskupení objektů zájmů (investiční programy, IT služby, IT projekty a další IT aktiva a zdroje), které jsou řízeny a monitorovány tak, aby optimalizovaly obchodní hodnotu. Investiční portfolio je primárním zájmem metodiky Val IT, zatímco IT služby, projekty, aktiva a zdroje jsou objektem zájmu metodiky Cobit [Svatá 2011].

Další významnou oblastí IT Governance je oblast řízení rizik. Řada mezinárodních standardů a metodik se často zabývá pouze business riziky a nejsou tak dostatečně relevantní pro oblast řízení rizik IT, nebo pokud se riziky IT zabývají, zužují tuto oblast na řízení rizik bezpečnosti a neodpovídají tak pojetí IT Governance. Dokument Risk IT definuje komplexní rámec pro řízení rizik, který odpovídá holistickému pojetí (integruje různé úrovně řízení rizik a současně je v přijatelném detailu aplikovatelný pro oblast řízení rizik IT. Společně s dokumenty Cobit a Val IT představují velmi provázaný model kontrol, investic a rizika pro oblast řízení a poskytování služeb IT. Risk IT se skládá ze dvou částí: The Risk IT Framework (Rámec pro řízení rizik IT), který vysvětluje základní principy řízení rizik procesní model, a The Risk Practitioner Guide (Praktický návod pro řízení rizik IT), který obsahuje praktické návody pro tuto oblast [Svatá 2011].

5.2 Obecný postup provádění auditu IS/IT

Cílem obecného postupu provádění auditu je zmapovat jeho etapy, činnosti a výstupy, které audit doprovázejí. Všechny etapy nemusejí být nutnou součástí každého auditu, některé se mohou vynechat nebo modifikovat podle objektu auditu a prostředí, ve kterém se realizuje. Souhrnný přehled jednotlivých etap, jejich vazeb na etapy, podle dokumentu IT Assurance Guide, činností a výstupů ukazuje Obrázek 21 [Svatá 2011]. Popis těchto etap následuje v dalších podkapitolách práce.

Etapy	Etapy podle IT Assurance Guide	Činnosti	Výstupy
0. Etapa: Uzavření smlouvy na audit	Plánování a určování rozsahu	Specifikace: <ul style="list-style-type: none"> ▪ předmětu auditu ▪ cílů auditu ▪ rozsahu auditu ▪ kritéria hodnocení ▪ organizace auditu ▪ harmonogram ▪ výstupů ▪ cena auditu ▪ obecných standardů (ochrana tajemství) 	Podepsaná smlouva na audit
1. Etapa: Předběžné plánování		<ul style="list-style-type: none"> ▪ Porozumění podnikatelským procesům ▪ Porozumění architektuře IS/IT ▪ Porozumění systému vnitřních kontrol 	<ul style="list-style-type: none"> ▪ Potvrzení správnosti porozumění ▪ Dokumentace auditu ▪ Dokumentace architektury IS/IT ▪ Program auditu
2. Etapa: Vytvoření plánu auditu	Testování návrhu kontrol (A)	<ul style="list-style-type: none"> ▪ Hodnocení kontrol ▪ Plán testů ▪ Určení potřebných zdrojů pro audit a jejich rozvrh 	Plán auditu
3. Etapa: Realizace auditu	Testování výstupu kontrol (B)	<ul style="list-style-type: none"> ▪ Realizace testů kontrol ▪ Realizace podrobných testů ▪ Analýza a vyhodnocování výsledků 	<ul style="list-style-type: none"> ▪ Výsledky testů ▪ Předběžná auditorská zpráva
4. Etapa: Závěr a vydání auditorské zprávy	Dokumentace dopadu slabín (C)	<ul style="list-style-type: none"> ▪ Rozhovor s managementem ▪ Zpracování auditorské zprávy a výroku 	Výsledná auditorská zpráva
5. Etapa: Sledování plnění závěrů auditorské zprávy		Vytvoření plánu sledování	Průběžná hlášení managementu

Obrázek 21: Obecný postup auditu. Zdroj [Svatá 2011].

5.2.1 Uzavření smlouvy na audit (0.Etapa)

Předmět auditu znamená vymezení objektu (prvku) informačního systému. V některých případech zadavatel auditu (externí) dokáže popsat, co by mělo být předmětem auditu (např. aplikace, proces, projekt apod.). V řadě případů externích auditů a téměř u každého interního tomu tak není, a potom musí auditor (s pomocí zadavatele) předmět

sám určit a ohraničit. Je to důležité zejména proto, aby byla před zahájením auditu jasná omezení a přínosy auditu a předešlo se tak nedorozumění v očekávání mezi zadavatelem a auditorem [Svatá 2011].

Při plánování může auditor postupovat několika různými způsoby, které by měl ideálně vzájemně kombinovat [Svatá 2011]:

- vertikální přístup upřednostňuje „postup shora dolů“ a auditor se nejdříve seznamuje se strategickými cíli podniku, vybírá ten nejdůležitější nebo nejrizikovější, cílům přiřazuje vhodné IT cíle a ty potom provazuje na vhodné IT procesy a jejich kontrolní cíle,
- horizontální přístup upřednostňuje pohled jednotlivých objektů/prvků IS, kterými mohou být aplikace, zařízení, lidé, které se potom sdružují např. podle business procesů, podle funkčních oblastí řízení podniku, prvků infrastruktury nebo dat,
- přístup využívající analýzu rizik vychází z mapování jednotlivých aktiv (prvků) IS, jejich slabin, hrozeb (velikost škody, pravděpodobnost realizace) a na jejich základě definuje předmět auditu,
- dynamický přístup upřednostňuje životní cyklus IS a jeho prvků od pořízení, testování, implementaci, provozování a monitorování až po vyřazení nebo zrušení.

Cíl auditu má úzkou vazbu na předmět auditu. Pokud se např. vytipuje, že předmětem auditu bude určitý proces, aplikace, projekt, potom je třeba upřesnit, jaký atribut nebo atributy tohoto prvku bude audit hodnotit. V zásadě může jít o účelnost, účinnost, bezpečnost, soulad s regulacemi a spolehlivost. Auditor může společně se zadavatelem a ve vazbě na konkrétní předmět auditu určit i jiné atributy. Měla by se však specifikovat jejich případné omezení. Upřesnění kritérií hodnocení je důležité pro zajištění objektivitu auditu. Pokud standardy pro daný předmět neexistují, potom by ve smlouvě měly být popsány mechanismy a postupy, kterými se vytvoří (např. rozhovory s manažery, které pomohou určit hranice a hodnoty metrik pro stanovení hranice toho, co je přijatelné a co nikoli) [Svatá 2011].

Rozsah, resp. hloubka auditu, určuje, zda se provedou všechny kroky etapy Realizace auditu. Jinými slovy jde o to, zda auditor provede pouze pozorování vůči nějakému vybranému kritériu hodnocení předmětu a cíle auditu. Výstupem potom bude

hodnocení toho, jak předmět splňuje náležitosti kritéria. Podrobnější rozsah auditu zahrnuje testování toho, jaké reálné výstupy předmět auditu poskytuje a co se s nimi dělá. Audit je pak náročnější a dražší. Nejpracnější je audit, který se mimo předchozí aktivity snaží nejlépe finančně vyjádřit, jaký vliv mají zjištěné nedostatky na cíle podnikání. Tyto dopady však často není možné finančně vyčíslit [Svatá 2011].

Organizace auditu je důležitá v těch případech, kdy jde o komplexnější podrobný audit a jeho realizaci bude zajišťovat více osob. Základním požadavkem je, aby byl auditor (event. celý tým) nezávislý na auditovaném objektu. Dále je vhodné určit: vedoucího auditorského týmu a jeho členy, místo v organizaci, kontaktní osobu z organizace zadavatele, pravidla pro komunikaci a prostředky, které bude auditor používat. Další důležitou součástí smlouvy je harmonogram auditu, protože vymezuje časový úsek, ve kterém se budou realizovat testy, na jejichž základě se budou formulovat závěry. Harmonogram by měl vycházet z nějaké metodiky, aby průběh auditu splňoval požadavek formalizovatelnosti [Svatá 2011].

Při popisu výstupů z auditu IS jsou důležité tyto informace [Svatá 2011]:

- počet (výstup musí být vždy minimálně jeden), pokud je výstupů více, mělo by být určeno, k jakým etapám se budou vázat,
- komu budou výstupy určeny,
- v jaké formě se adresátům dodají a jakými kanály,
- zda se zpracují dvě verze zpráv: jedna pro manažery, druhá pro IT specialisty,
- jaký bude stupeň utajení výstupů,
- zda bude součástí výstupů i jejich prezentace,
- do jaké doby má auditovaný subjekt nebo zadavatel auditu reagovat s připomínkami k výstupům auditu.

Cena auditu může být stanovena po dohodě s klientem jako pevná, nebo může být odvozena od hodinových sazeb podle ceníku služeb. Na výši ceny má často vliv hodnota nebo důležitost auditovaného předmětu. Kromě těchto klíčových částí smlouvy by konkrétní smlouva na provedení auditu měla ještě obsahovat další části, jako např.: práva auditovaného, právo auditora na přístup k informacím (SW, zařízení, atd.), prohlášení auditora o zachování mlčenlivosti a další části (sankce, penále, způsob ukončení smlouvy apod.) [Svatá 2011].

5.2.2 Předběžné plánování (1.Etapa)

Pro plánování činnosti potřebuje auditor obvykle porozumět podnikovým procesům, architektuře IS/IT a systému vnitřních kontrol. Ke splnění tohoto úkolu musí získat řadu informací studiem dokumentace, pozorováním v pracovním prostředí, dotazníky nebo interview. Míra porozumění je dána cílem auditu. Čím obecnější je cíl, tím je třeba komplexnější porozumění podnikovým procesům. Porozumět business procesům v organizaci je snazší tam, kde existuje mapa procesů a procesy jsou modelovány ať už slovně, nebo nástroji pro modelování (např. *Business Proces Management*). Rovněž porozumění architektuře IS je pro audit IS velmi důležité, zvláště ve velkých organizacích. Vnitřní kontrolní systém by měl být předmětem každého auditu. Je totiž rozdíl v hodnocení toho, zda systém kontrol existuje a hodnocení toho, zda je efektivní [Svatá 2011].

Často se stává, že požadovaná dokumentace není v organizaci k dispozici, nebo není v aktuálním stavu. Pak by měl auditor takovou dokumentaci společně s odpovědnými pracovníky vytvořit, což může znamenat zpoždění auditu, případně dodatek ke smlouvě. Tento výstup je však přínosem pro zadavatele ještě před zahájením vlastního auditu. Výstupy z této etapy jsou: potvrzení správnosti porozumění, dokumentace auditu, dokumentace architektury IS a program auditu [Svatá 2011].

Dokumentace auditu zahrnuje jednak popis práce a výstupů auditora a dále evidenci všech informací, které byly podkladem pro nálezy a závěry auditora. Dokumentace by měla být úplná, srozumitelná a vhodně strukturovaná. Lze ji využívat např. pro: doložení míry souladu auditu s existujícími standardy, doložení míry plnění smlouvy, podpora dalších aktivit, apod. Měla by minimálně zahrnovat [Svatá 2011]:

- dokumentaci z předchozího auditu,
- plánování a přípravu předmětu, cíle, a rozsahu auditu,
- přehled schůzek souvisejících s auditem a délku jejich trvání,
- program a postupy auditu,
- nálezy, závěry a doporučení auditu,
- výstupy auditu.

Důležitými náležitostmi dokumentace auditu jsou informace o zdroji a datu jejich získání. Vzhledem k tomu, že dokumentace auditu často může obsahovat velmi citlivé nebo tajné informace, musí se v organizaci vytvořit taková pravidla, která určují způsob

ukládání a nakládání s touto dokumentací tak, aby byly dodrženy právní a profesní požadavky stejně jako požadavky zadavatele auditu [Svatá 2011].

5.2.3 Vytvoření plánu auditu (2.Etapa)

Na základě odsouhlasených výstupů předchozí etapy auditor provádí analýzu získaných a ověřených informací. Využívá své zkušenosti i existující standardy (best-practice). Oblasti, které se ukáží být pro podnik významné a zranitelné, dále auditor podrobí analýze efektivnosti a adekvátnosti návrhu jejich kontrol. Pokud se ukáže, že kontroly nejsou efektivní, nebo dokonce neexistují, není třeba je testovat a v dalším postupu se auditor zaměří na návrh nové koncepce těchto kontrol. Pokud se ukáží kontroly jako efektivní, pak je třeba rozhodnout o pokračování jejich testování, tzn. bude se ověřovat, jak se realizují v praxi a jak se nakládá s jejich výstupy [Svatá 2011].

Hlavním výstupem z etapy je upřesněný plán auditu, zahrnující [Svatá 2011]:

- upřesnění cílů auditu (předmětu auditu),
- výsledky hodnocení kontrol a zdůvodnění dalšího postupu auditu,
- plán testů,
- určení potřebných zdrojů pro audit a jejich rozvrh.

5.2.4 Realizace auditu (3.Etapa)

Vlastní realizace auditu spočívá v testování kontrol, které byly identifikovány a analyzovány jako efektivní. Jde o testování souladu popisu/záměru kontrol s jejich praktickou realizací. V případě, že testy prokáží nesoulad, je třeba zjistit míru škody, která byla tímto nesouladem způsobena. K tomu slouží tzv. substantivní testy, jejichž předmětem jsou již jednotlivé transakce a nikoliv kontroly. Pokud toto testování není vzhledem k rozsahu reálné, je možné použít různé metody výběru vzorků. O realizaci a výsledcích testů je třeba vést důkazní materiál, který je součástí dokumentace auditu. Základními výstupy z této etapy jsou výsledky jednotlivých testů a předběžná auditorská zpráva [Svatá 2011].

Nejčastější druhy důkazního materiálu jsou tyto: pozorované procesy a existence fyzických položek, dokumentace (papírová, na médiích), ověřená tvrzení auditovaných, analýzy informací. Pro získávání důkazů je možné použít různé postupy – takové, které nejméně zatěžují auditovaného. Je doporučeno začít studiem dokumentace, pozorováním prostředí, dotazníky, interview. Jako významný zdroj důkazů mohou

sloužit různé automatizované kontroly (reporty, logy, které si auditor může ve spolupráci s administrátory nastavit [Svatá 2011]).

5.2.5 Závěr a vydání auditorské zprávy (4.Etapa)

Předběžná auditorská zpráva by měla být projednána s odpovědnými pracovníky, jichž se předmět auditu týká, případně se zadavatelem auditu. Hlavním smyslem tohoto předběžného projednání auditorské zprávy je [Svatá 2011]:

- ujištění auditora, že jeho nálezy a závěry jsou relevantní a nebyla pominuta hlavní rizika,
- poskytnutí možnosti auditovaným provést rychlou nápravu u méně podstatných nálezů a tím redukovat konečnou zprávu na skutečně závažné problémy.

Po předběžném projednání je možné vytvořit konečnou verzi auditorské zprávy, která je hlavním výstupem této etapy.

5.2.6 Sledování plnění závěrů auditorské zprávy (5.Etapa)

U externího auditu je tato etapa problematická, protože jde obvykle o jednorázový audit, který končí prezentací a projednáním konečné auditorské zprávy. Auditor pak nemá možnost kontrolovat, zda se organizace řídí doporučením auditora. Doporučení auditora mohou být přitom různě náročná na zdroje, a tedy by auditor měl vytvořit plán sledování jejich reálného plnění. Výstupem z etapy mohou být průběžná hlášení auditora, jak se doporučení plní v praxi [Svatá 2011].

6 Koncept Quality Measure Elements

V této kapitole je detailně popsán princip prvků měření kvality (*Quality Measure Elements*) vycházející z návrhu mezinárodní normy [ISO/IEC 2011a], která rovněž sloužila jako zdroj pro texty uvedené dále v podkapitolách. Tyto prvky mají být použity při hodnocení jakosti IS během životního cyklu SW produktu.

Tento standard zahrnuje soubor fundamentálních měř, které lze na produktu či na jeho funkci měřit přímo, tedy pouze pozorováním produktu nebo jeho chování (funkce), a které slouží za základ pro výpočet jednotlivých měř jakosti. Tato primitiva jsou důležitá nejen pro hodnocení jakosti, ale i pro stanovení dalších významných charakteristik softwarového inženýrství, jako je tak zvaná vývojová složitost softwaru a z ní vyplývající nároky na pracnost realizace, odhady minimální „rozumné“ doby potřebné na realizaci projektu a podobné velmi důležité charakteristiky [Vaníček 2004b].

6.1 Prezentace modelu metody měření

Aby bylo možné pochopit a indikovat charakteristiky a podcharakteristiky jakosti, je vymezeno měření kvality QM. Pak jsou na základě měření kvality definovány QME. Měření je aplikováno na QME pro vytvoření QM. Metoda měření se použije na vlastnost definovat a určit způsob, jak kvantifikovat QME.



Obrázek 22: Vztah mezi kvantifikovanou vlastností, metodou měření a QME.

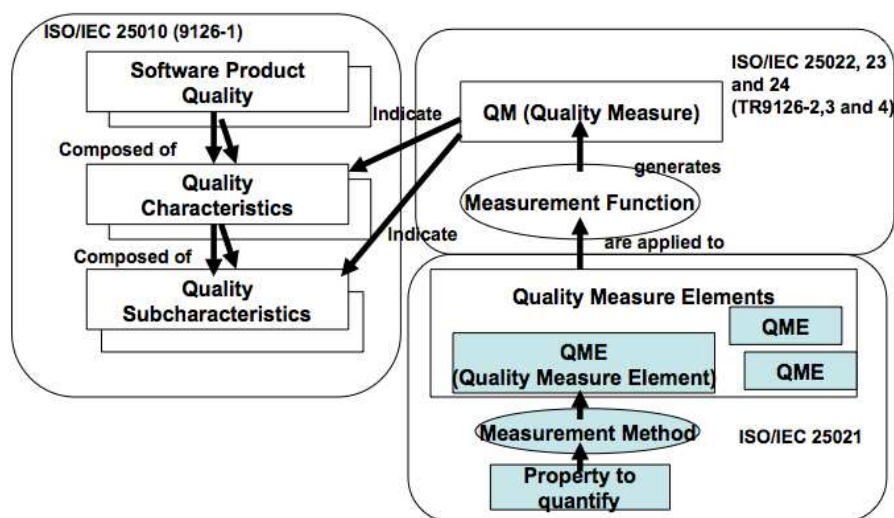
Zdroj [ISO/IEC 2011a].

Uživatel metody měření stanoví a shromáždí data související s kvantifikací vlastnosti (obr. 22). V závislosti na cíli a kontextu použití QME, může řadu vlastností a subvlastností určit. Jde o vstupy metody měření. Tyto vlastnosti jsou extrahovány a definovány ze software produktu (např. dokumentace, kód). Uživatel metody měření

vytváří různé výstupy, které jsou identifikací vlastností pro výsledné určení číselného přiřazení pravidel.

Obrázek 23 ukazuje, že:

- kvalita produktů se skládá z charakteristik jakosti, které pak mohou být složeny z podcharakteristik,
- míry jakosti produktů se používají k indikaci charakteristik a podcharakteristik jakosti,
- vztah mezi vlastností ke kvantifikaci, metodou měření a QME.



Obrázek 23: Vztah mezi vlastností ke kvantifikaci, metodou měření, QME a metodou měření.

Zdroj [ISO/IEC 2011a].

6.2 Základní sada QME

Různé QME mohou být použity společně a v kombinaci definovat měření jakosti. Některé QME pocházejí z ISO/IEC 9126 a další jsou z potřeb průmyslového trhu a existující normy, jako je měření funkční velikosti. QME uvedené v tomto počátečním setu se týkají (sub) charakteristik jakosti modelu kvality produktů uvedeného v ISO/IEC 25010. Počáteční sada QME je navržena pro uživatele tohoto dokumentu pro zvážení její použitelnosti, když se připravuje měření jakosti produktu.

a) Název QME	QME musí mít jedinečný název a musí být označena v případě potřeby pořadovým číslem. Většinou začíná jako "počet z ... (měřicí stupnice)."
b) Cílová entita	QME musí mít cíl, který má být charakterizován měřením jeho vlastnosti. Cílový subjekt by měl být vyvíjený produkt nebo chování systému, software, nebo zúčastněné strany, jako jsou uživatelé, provozovatelé, vývojáři, testéři, nebo správci.
c) Cíle a vlastnosti ke	Identifikace vlastnosti ke kvantifikaci obvykle souvisí s názvem QME. Vlastnost pro cílové entity pro QME (Příklad: QME = počet chyb, cílová vlastnost = poruchy).

kvantifikaci	Vybraná vlastnost ke kvantifikaci by měla být ta, která je nejdůležitější pro měření potřebné informace. Daná vlastnost může být zahrnuta do několika konceptů měření. Příklad: "Počet poruch software" je QME a "porucha" je vlastnost software ke kvantifikaci.
d) Relevantní míra(y) kvality	Odkaz na specifické míry kvality, které používají QME, musí být uveden. Například, míry kvality v ISO/IEC 9126 sérii, 25000 SQuaRE sérii a v dalších dokumentech.
e) Metoda měření	Metoda měření umožňuje, jak shromažďovat a jak transformovat na hodnotu kvantifikovanou vlastnost dle matematické funkce. Následující informace: kontext QME, proces životního cyklu software, omezení měření a numerická pravidla jsou součástí metody měření.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci (volitelně)	Identifikovaná vlastnost ke kvantifikaci může být v souvislosti s některými sub-vlastnostmi, pokud je třeba. Tento vztah mezi vlastnostmi by měl být vyjádřen jako schéma nebo vzorec. To představuje model metody měření. Například, v rámci metody COSMIC, funkční proces je vlastnost, která může být vyjádřena v modelu některými sub-vlastnostmi, jako jsou vstup, čtení, zápis a konec. To může pomoci identifikovat vlastnost ke kvantifikaci „přesun dat“, která je relevantní pro funkční velikost vycházející z měření.
g) Vymezení dílčích vlastností (volitelně)	Existuje-li seznam dílčích vlastností, měla by být každá dílčí vlastnost definována.
h) Vstup pro QME	Vstup musí být popsán dostatečně podrobně pro identifikaci, jaké kvantitativní informace se použijí k měření QME. Jakékoli zdroje poskytující vstup by také měly být označeny jako produkty, chování systému a softwaru, nebo lidského chování uživatelů, operátorů, vývojářů, testerů, nebo správců. Pak mohou být vstupy sub-vlastností nebo kvantitativní informace, které se jich týkají.
i) Jednotka měření pro QME	Pro měrnou jednotkou, pokud je to vhodné, se používá vzorec. Příklad jednotek zahrnují počet X, procentní podíl a pozice.
j) Numerická pravidla	Stanovení numerických pravidel musí být popsáno z praktického pohledu (obecně v textové podobě), nebo z teoretického hlediska (obecně matematického vyjádření). Vnitřní konzistence je často problém při přiřazování numerických pravidel. Je důležité zachovat konzistenci mezi vlastností a sub-vlastností, které je třeba měřit. Například, při měření poruch se uvede počet poruch. Ale pokud se rozlišuje mezi hlavními a menšími poruchami, bude přesnější měření získáno přidáním zvlášť velkých a malých chyb. Interpretace musí brát v úvahu omezení výsledku vztahující se na každou vlastnost a sub vlastnosti.
k) Typ měřicí stupnice	Typ měřicí stupnice musí být identifikován. Může být nominální, ordinální, intervalová nebo poměrová.
l) Kontext QME	Poskytuje informace o plánovaném využití výsledků měření. Je užitečné popsat typické příklady charakteristik jakosti, sub-charakteristik jakosti nebo měření kvality (QM), které jsou určeny hlavně pro použití QME výsledků měření. Předpoklady a nezbytné podmínky cílových entit, jejich prostředí a okolnosti, které metoda měření QME umožňuje použít, by měly být popsány.
m) Software životního cyklu procesu (ů)	Typický vhodný životní cyklus procesu(ů) by měl být určen zde pro každý QME, který vytváří nebo realizuje konkrétní cílové entity umožňující získat aktuální naměřené hodnoty QME.
n) Omezení měření (volitelně)	Jakákoli omezení týkající se metody měření je třeba popsat, je-li to nutné. Měření QME mohou mít omezení, jako chybu měření nebo odchylku vzniklé v důsledku závislosti na následujícím: rozsah šetření, způsob šetření, nestabilní specifikace a taktiky testovacích případů.

Tabulka 2: Formát metody měření pro QME. Zdroj: [ISO/IEC 2011a].

Dále je uvedena základní sada prvků měření kvality, uvedená v mezinárodním standardu [ISO/IEC 2011a]. Každá položka v níže uvedené tabulce je rozdělena do čtyř skupin tak, že "a)" pro QME identifikaci, "b) - d)" co je QME, "e) - k)", jak měřit QME a "l) - n)" řízení aplikace QME.

#	Základní sada QME	Definice a pojetí vztahující se přímo k QME
1	Počet přístupů k pomocným funkcím	Přístupnost: použitelnost výrobku, obsluha, prostředí nebo možnosti podle lidí s nejširším rozsahem schopností.
2	Počet uživatelských problémů	Každá reklamace podaná uživatelem na produkt je organizací registrována (obvykle na úrovni helpdesku). Znalost stížností by mohla pomoci měřit spokojenosti uživatelů v průběhu časového období. Například: technické problémy nebo funkční problémy získané od uživatele reklamací a řešené helpdeskem.
3	Počet záznamů	Záznam: sada souvisejících datových položek považovaná za jednotku.
4	Doba trvání	Doba trvání: Celkový počet délky pracovní doby (bez svátků či jiné mimopracovní doby), potřebné k dokončení harmonogramu aktivit nebo komponent struktury rozpisu práce. Obvykle vyjadřováno jako pracovní dny nebo pracovní týdny.
5	Úsilí (v jednotce času)	Úsilí: Počet pracovních jednotek potřebných k dokončení harmonogramu aktivit nebo komponent struktury rozpisu práce. Obvykle vyjadřováno jako pracovní hodiny, dny, týdny.
6	Počet poruch systému	Selhání systému: Kompletní systém zahrnuje všechny související zařízení, materiál, počítačové programy, firmware, technickou dokumentaci, služby a personál potřebný pro provoz a podporu v nezbytné míře pro soběstačné použití v daném prostředí v předem stanovených mezích. Selhání software: ukončení schopnosti produktu plnit požadované funkce nebo neschopnost je plnit v předem stanovených mezích.
7	Počet poruch	Chyba: 1. ukončení schopnosti výrobku plnit požadované funkce nebo neschopnost plnit je v předem stanovených mezích. 2. událost, při které systém nebo součást systému neplní požadovanou funkci v rámci stanovených mezí.
8-1	Počet poruch (kód)	Chyba: nesprávný krok procesu, nebo definice dat v softwarovém kódu. Poznámka: chyba, pokud se objeví, může způsobit poruchu.
8-2	Počet závad (design)	Chyba: nesprávný krok procesu nebo definice dat ve specifikacích návrhu software.
8-3	Počet závad (požadavky)	Chyba: nesprávný krok procesu, definice dat v požadavcích na software.
9	Funkční velikost produktu	Funkční velikost: velikost software odvozená od kvantifikace funkčních požadavků uživatelů.
10	Počet přerušení	Přerušení: pozastavení procesu řízené externí událostí procesu.
11	Počet datových položek	Datové položky: nejmenší zjistitelná jednotka dat v určitém kontextu, pro které jsou definice, identifikace, přípustné hodnoty a další informace uvedené formou souboru vlastností.
12	Počet chybových hlášení	Chybová zpráva: zpráva, kterou dá aplikace, pokud jsou zadány nesprávné údaje, nebo když se vyskytne další procesní chyba.
13	Počet chyb	Chyba: 1.lidská činnost, která vede k nesprávným výsledkům, jako je například software obsahující chybu. 2. nesprávný krok procesu nebo

		definice dat. 3. nesprávný výsledek. 4. rozdíl mezi vypočtem, pozorováním, nebo měřením hodnot nebo stavu a skutečnými, určenými, nebo teoreticky správnými hodnotami nebo podmínkami.
14	Počet zpráv	Zpráva: Informace poskytnuté koncovým uživatelům softwarových systémů pro informace, řízení, výstražné účely.
15	Počet kroků (procedur)	Krok: 1. Jeden prvek (číslovaný v seznamu položek) v postupu, který říká uživateli provedení akce (nebo akcí). 2. Současný výskyt konečné množiny z přechodových režimů, které jsou zároveň povoleny označit. 3. Abstrakce akce používané v procesu, která může zanechat nespecifikované objekty, které se účastní této akce. Poznámka: krok obsahuje jednu nebo více akcí. Response software, nejsou považovány za kroky.
16	Počet úkolů	Úkol: 1. požadované, doporučené nebo povolené akce, s cílem přispět k dosažení jednoho nebo více výstupů procesu. 2. v designu software, komponentě software, která může pracovat paralelně s ostatními softwarovými komponentami. 3. činnosti potřebné k dosažení cíle. 4. současně objekt s vlastním podprocesem kontroly. 5. posloupnosti instrukcí považované za základní jednotku práce podle supervize operačního systému. 6. nejmenší jednotka práce podléhající řízení odpovědnosti, dobře definovaný pracovní úkol pro jednoho nebo více členů projektu.
17	Počet testovacích případů	Testovací případ: minimální samostatně spustitelné části testovací sady softwarového systému, které dají 2 možné výsledky [fail, pass].
18	Počet use cases	Use case je popis interakce mezi Actorem (iniciátor interakce), a systémem jako takovým. To je reprezentováno jako sled jednoduchých kroků. Tato vlastnost je funkční velikost, ale z pohledu use case.
19	Počet operací	Pokračující provádění činností, které produkují stejný výrobek nebo poskytují službu opakovaně. Provoz: 1. proces běhu počítačového systému v zamýšleném prostředí pro výkon jeho zamýšlené funkce. 2. aktivity nutné k provedení činnosti.
20	Počet kritických chyb	Fatal error: chyba, která vyústí v úplnou neschopnost funkce systému nebo jeho součásti.
21	Velikost databáze	Počet výskytů v databázi. Databáze: 1. soubor vzájemně souvisejících dat uložených v jednom nebo více počítačových souborů. 2. sběr dat organizovaných podle konceptuální struktury popisující vlastnosti dat a vztahy mezi jejich příslušnými entitami, podpora jedné nebo více aplikačních oblastí. 3. sběr dat popisujících konkrétní cílovou oblast, která je používá a aktualizuje jednu nebo více aplikací.
22	Velikost paměti	Množství počítačů nebo zařízení na ukládání (vyjádřeno v násobcích byte). Paměť: adresovatelná místa v procesoru a další vnitřní paměť, která slouží ke spuštění instrukcí. Kapacita paměti: maximální počet položek, které mohou být drženy v dané paměti počítače, obvykle se měří ve slovech nebo bytech.

Tabulka 3: Základní set QME. Zdroj: [ISO/IEC 2011a].

V následujících tabulkách je definován a kvantifikován počáteční soubor QME podle tabulky 3. Popis procesu návrhu QME uvedený v kapitole 6.3 lze použít ve spojení s tabulkou 3 k definování a kvantifikaci jednotlivých QME, nebo pomoci definovat nové QME v organizaci. Tyto tabulky nezobrazují kompletní mapování mezi QME

a jejich použití pro charakteristiky jakosti, zejména podcharakteristiky. Tento počáteční set je možné v průběhu času změnit.

Characteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost						Účinnost		
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
1-Počet přístupů k pomocným funkcím											X			
2-Počet uživatelských problémů	X		X			X	X	X	X	X	X			
3-Počet záznamů											X			X
4-Doba trvání				X	X			X		X		X		
5-Úsilí (v jednotce času)							X	X					X	
6-Počet poruch systému							X	X	X					
7-Počet poruch		X							X					
8-Počet závad									X					
9-Funkční velikost produktu	X												X	
10-Počet přerušení	X													
11-Počet datových položek											X			
12-Počet chybových hlášení											X			X
13-Počet chyb		X							X					
14-Počet zpráv											X			
15-Počet kroků (procedur)			X					X						
16-Počet úkolů	X		X			X			X					
17-Počet testovacích případů								X	X		X			

Charakteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost					Účinnost			
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
18-Počet use cases	X						X	X	X	X	X			
19-Počet operací	X						X	X	X	X	X			
20-Počet kritických chyb		X								X				
21-Velikost databáze														X
22-Velikost paměti														X

Tabulka 4: mapování mezi počáteční sadou QME a jejich použitím pro Funkční přiměřenost, Kompatibilitu, Použitelnost a Účinnost dle ISO/IEC 25010. Zdroj: [ISO/IEC 2011a].

Charakteristiky jakosti	Bezporuchovost				Bezpečnost					Udržovatelnost				Přenositelnost			
	Zralost	Dostupnost	Odolnost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
1-Počet přístupů k pomocným funkcím																	
2-Počet uživatelských problémů	X	X	X	X	X	X	X	X	X						X	X	X
3-Počet záznamů				X	X	X	X	X	X						X		
4-Doba trvání												X	X	X			
5-Úsilí (v jednotce času)												X	X	X			
6-Počet poruch systému												X	X	X			
7-Počet poruch	X	X	X	X						X	X	X	X	X			
8-Počet závad	X	X	X	X						X	X	X	X	X			
9-Funkční velikost	X	X	X	X													

Charakteristiky jakosti	Bezporuchovost				Bezpečnost					Udržovatelnost					Přenositelnost		
	Zralost	Dostupnost	Odolnost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
produktu																	
10-Počet přerušení														X			
11-Počet datových položek				X	X	X	X	X	X						X		
12-Počet chybových hlášení		X	X												X		
13-Počet chyb	X	X	X	X													
14-Počet zpráv																	
15-Počet kroků (procedur)																	
16-Počet úkolů															X		
17-Počet testovacích případů													X				
18-Počet use cases			X	X	X	X	X	X	X								
19-Počet operací			X	X	X	X	X	X	X								
20-Počet kritických chyb	X	X	X	X									X				
21-Velikost databáze																	
22-Velikost paměti																	

Tabulka 5: mapování mezi počáteční sadou QME a jejich použitím pro Bezporuchovost, Bezpečnost, Udržovatelnost a Přenositelnost dle ISO/IEC 25010. Zdroj: [ISO/IEC 2011a].

6.3 Projektování Prvků měření kvality (QME)

Tento dokument [ISO/IEC 2011a] stanoví postup aplikace metody měření, jak je doporučeno v ISO/IEC 15939. Následující podkapitoly z tohoto standardu [ISO/IEC 2011a] vycházejí.

Metoda měření kvality generuje Prvek měření kvality (QME) z vlastnosti ke kvantifikaci. Záměrem je pomoci uživatelům ISO/IEC 9126 série (ISO/IEC TR 9126-2, ISO/IEC TR 9126-3, ISO/IEC TR 9126-4) a uživatelům řady norem měření kvality

SQuaRE (ISO/IEC 25020, ISO/IEC 25022, ISO/IEC 25023 a ISO/IEC 25024) aplikovat metody měření. Tato příloha bude užitečná při výběru a používání různých měření jakosti pro hodnocení kvality produktu v rámci celého životního cyklu.

Jak ukazuje obrázek 22. (kvantifikace QME), tvůrce metody měření identifikuje a shromáždí údaje týkající se kvantifikace vlastností QME. V závislosti na kontextu použití a cílů QME lze určit řadu dílčích vlastností. Jde o vstup pro metody měření. Tyto vlastnosti jsou extrahovány a definovány ze software (např. životního cyklu software). Tvůrce metody měření vytváří různé výstupy, které identifikují vlastností a sub-vlastností. Tvůrce konstruuje princip měření a popis metody měření pro implementaci přiřazení numerických pravidel.

Tato část popisuje postup (jednotlivé kroky) pro tvorbu metody měření, od identifikace QME po číselné přiřazení (jednotka měření). Následující část obsahuje příklady QME vyplývající z použití tohoto postupu měření.

Navrhované kroky návrhu QME:

1. identifikace QME a cíle,
2. identifikace vlastnosti ke kvantifikaci související s QME,
3. vymezení vlastnosti a sub-vlastností,
4. konstrukce modelu vlastností, které mají být kvantifikovány,
5. přiřazení jednotky měření (vzorec) a typ měřicí stupnice.

Teoreticky mohou být QME aplikovány na jakékoli měření kvality a kdykoli během celého životního cyklu produktu. V tomto dokumentu se návrh procesu vztahuje na implementaci metody měření a je nezávislý na QME a technologii. Je však možné definovat úvahy týkající se cílů měřením při návrhu konkrétních QME. To je nutné proto, že konkrétní QME souvisí s měřením jakosti, které souvisí s podcharakteristikami. To nevyklučuje použití QME pro různé charakteristiky a podcharakteristiky, pokud se cíle měření nemění.

6.3.1 Identifikace QME a cílů

Seznam QME byl extrahován z měření kvality v ISO/IEC 9126 sérii, část 2, 3 a 4. Pro každý uvedený QME byla identifikována vlastnost (produktu). Tato práce identifikovat QME znamená průběžnou aktualizaci, protože to závisí i na nových identifikovaných charakteristikách a podcharakteristikách (viz ISO/IEC 25010), které přinesou také nové "míry kvality" a možné nové QME se svými příslušnými vlastnostmi. Konečně,

identifikace QME v souvislosti s jejich využitím je důležitá, protože poskytuje informace o cílech měření a zamýšleného použití výsledků měření. V případě, že identifikujeme QME, je možné určit podle QME stálou vlastnost používání.

V důsledku toho by měl návrh měření obsahovat následující deskriptivní údaje: jméno QME, název vlastnosti ke kvantifikaci, název zvolené metody měření, kontext měření (použití podle měření jakosti čeho), životní cyklus software, omezení, úhel pohledu, definice vlastnosti ke kvantifikaci a reference, seznam sub-vlastností a reference, vztahy mezi vlastnostmi a sub-vlastnostmi, zdroje informací, vstup pro měření, jednotku QME, numerická pravidla pro QME a typ měřicí stupnice. Pokud jde o cíle, designer by měl také objasnit, zda měření vlastnosti bude provedeno například z hlediska uživatele nebo vývojáře. V rámci ISO/IEC 9126 jsou k dispozici tři hlediska: vnitřní (developer), externí (uživatel) a jakost užití (při používání software uživatelem). Dokument by měl vyjasnit, v jakém životním cyklu vývoje software je nejlepší použít metodu měření.

6.3.2 Identifikace vlastnosti ke kvantifikaci v souvislosti s QME

Software je nehmotný produkt, ale přesto může být viditelný prostřednictvím více reprezentací. Sada obrazovek a zpráv pro uživatele, sada řádků kódu pro programátora, soubor reprezentací modelu software pro designera jsou dobrými příklady prvků software. Vlastnosti ke kvantifikaci v souvislosti s QME, osoba provádějící měření může vzít v úvahu existenci těchto prvků. Jednou z hlavních vlastností ke kvantifikaci je souvislost s jedním QME. Například, počet chyb je QME, zatímco "chyba" je hlavní vlastností ke kvantifikaci.

Pro organizace bude výběr vlastnosti v přímé souvislosti s výběrem QME. Výběr QME by se měl vztahovat k účelu programu měření v organizaci (Jaké charakteristiky a podcharakteristiky organizace chce kvantifikovat?). Každý QME použitý v organizaci může být organizací definován, pokud již není vymezen v rámci 25021.

Tabulka, která obsahuje návrh metody měření pro konkrétní QME by také měla poskytnout reference pro definice a návrh, pokud existují. Lze vycházet z existujícího standardu nebo publikací autorů. Pro každou identifikovanou vlastnost lze určit to, co je třeba měřit (např. velikost, atd.). Identifikovaná vlastnost se vztahuje k jedné nebo více

sub-vlastností, která může mít následovně dvě nebo více sub-vlastnosti. Konstrukce modelu je pak nezbytná.

6.3.3 Definice vlastnosti a sub-vlastností

Zjištěné vlastnosti ke kvantifikaci QME lze rozložit na měřitelné sub-vlastnosti. Například vlastnost velikosti "Use Case" lze rozložit na tři sub-vlastnosti: "Základní scénář", "Alternativní cesty" a "Výjimky". Autor metody měření by měl pak v odborné literatuře zjistit, jak jsou vlastnosti vztahující se k QME definovány a měřeny v předchozích výzkumech. Dále by měl sledovat podobnosti a rozdíly mezi definicemi vlastností v modelu kvality a dalších bibliografické odkazy. To záleží hlavně na cíli a kontextu užití QME. Výsledky hodnocení by měly být v souladu s cíli a použitím QME.

Popis takových vlastností lze provést v první fázi implicitně podle toho, jak jsou rozloženy do sub-vlastností. Tento rozklad popisuje, jakou roli hraje každá sub-vlastnost ve složení vlastnosti. Z tohoto důvodu, jak jsou vlastnosti rozkládány, by měly být popsány v tomto kroku.

6.3.4 Stavba modelu vlastnosti ke kvantifikaci

Vlastnost ke kvantifikaci QME je použita k získání vlastnosti(i) v modelu. Vztahy mezi definovanou (ými) vlastností (mi) nebo sub-vlastnosti, které představují software nebo jeho část, tvoří model. Model popisuje, jak rozpoznat vlastnost (i) a/nebo dílčí vlastnosti metody měření.

Zdroje dat použitých v metodě měření by měly být identifikovány v tomto kroku. Například prvky "dokument specifikace požadavků", "dokument popis testování" a další, poskytují důležité informace, které pomohou najít měřitelné vlastnosti.

Zdroj vstupních dat, který se používá ke kvantifikaci QME, by měl být určen. Například by mohl být součástí textu, ze kterého by bylo možné extrahovat potřebné informace pro kvantifikaci QME. Metoda měření pro QME může zahrnovat lidský úsudek (příklad: manuální zjišťování počtu závad) nebo nástroj (příklad: počítání počtu poruch po automatizovaném testu).

6.3.5 Přiřazení jednotky měření (vzorec) a typ měřicí stupnice

Přiřazení numerických pravidel je součástí návrhu procesu. Může být popsáno z praktického pohledu (zpravidla text), nebo z teoretického hlediska (obecně matematického výrazu). Vnitřní konzistence je často při přiřazování číselných pravidel

problém. Je důležité zachovat konzistenci mezi dvěma vlastnostmi, které je třeba měřit. Z tohoto důvodu je důležité ukázat, že při přidávání dvou vlastností (nebo subvlastností) jsou tyto spojeny společnou vlastností.

Interpretace je také spojena s druhem měřicí stupnice hodnot a matematických vztahů mezi hodnotami. Pokud ne, interpretace může být chybná. Je-li stupnice ordinálního typu, interpretace se týká pouze nižší hodnoty nebo vyšší hodnoty ve vztahu ke dvěma výsledkům. Spokojenost uživatelů 3 z 5 je nižší za předpokladu, že 1 je nejnižší a 5 nejvyšší.

7 Návrh metody pro kvantifikovaný přístup k řízení jakosti IS

V této kapitole bude detailně popsán návrh vytvoření metody a měr (QME) pro kvantifikovaný přístup k řízení jakosti. Návrh vychází z důkladné rešerše současného stavu problematiky a identifikovaných mezer. Na základě těchto zjištění z teoretické části práce, formulovaných předpokladů, i zkušeností autora, které jsou uvedené v následujících kapitolách práce, je dále navržena metoda specifikace kvantifikovaného přístupu k jakosti informačního zabezpečení. Rovněž je popsáno ověření navržené metody v praxi a výstup (výsledek) této metody. Nejprve je však nutné vymezit a charakterizovat informační zabezpečení podniků, pro tuto práci konkrétně bank působících na českém trhu.

7.1 Informační zabezpečení podniků

Podniková informatika je z obecného hlediska detailně popsána například v [Gála et al. 2009]. Informační zabezpečení podniků je v rámci této práce chápáno jako komplexní softwarové vybavení pro fungování společnosti, nikoli pouze bezpečnost (*security*) softwarových produktů. Pokud jde o informační zabezpečení v bankovním sektoru, pravděpodobně nejvýstižnější charakteristiku autor shledal v [INFO 2008], viz další odstavce této kapitoly.

Povaha finančního prostředí je natolik specifická, že vyžaduje také specifické softwarové nástroje a technologie, které není možné získat ve formě globálně nabízených softwarových aplikací. Ty v tomto případě nahrazují informační systémy vyvíjené na klíč. Informační systémy vyvíjené na klíč představují unikátní řešení odrážející komplexní nebo i konkrétně definované know-how podniku. U zakázkového vývoje softwaru je téměř jisté, že bude v cílovém prostředí reálně použitelný, neboť by měl být vytvářen na základě důkladné analýzy potřeb zadavatele. U sériově vyráběných aplikací to není zaručené téměř nikdy, protože do těchto produktů je implementována obecná, nikoliv přesně definovaná problematika bez ohledu na odvětví, kde bude produkt užíván. Neméně důležitým rysem zakázkového softwaru je jeho otevřenost pro další rozvoj. Tato vlastnost umožňuje relativně snadnou integraci nových funkčních požadavků a v konečném důsledku přináší značnou úsporu finančních prostředků, jež by jinak musely být investovány do zcela nových řešení. Pro oblast bankovníctví je vyvíjena celá řada specializovaných zakázkových softwarových řešení – ať už jde

o jednodušší aplikace podpůrného charakteru, nebo o rozsáhlé sofistikované informační systémy určené k řízení klíčových firemních procesů.

V [INFO 2008] je uvedeno konstatování, že v bankách prakticky neexistuje samostatný a ucelený informační systém řešící komplexní bankovní problematiku se všemi jejími souvisejícími oblastmi. S tímto názorem lze souhlasit, je však třeba doplnit, že zmíněná ucelenost se týká konkrétního informačního systému, který je hotovým a funkčním produktem, a komplexností je míněna provázanost na další systémy a aplikace, které tvoří informační zabezpečení banky.

Bankovní informační systém lze charakterizovat jako desítky různých, vzájemně provázaných informačních systémů a aplikací. Srdcem každého bankovního systému je primární úložiště dat, tzv. mainframe, který představuje hlavní účetní systém banky. Do tohoto “srdce” proudí veškerá data z okolních informačních systémů, které jsou na něj napojeny. Okolím mainframu je zpravidla tvořeno systémy, které podporují pracovní procesy prováděné na bankovních přepážkách, tzv. Front-Office systémy, dále aplikacemi zajišťujícími zpracování informací, tzv. Back-Office systémy, a ostatními podpůrnými systémy, které řeší administrativně-technickou problematiku. Systémy Front-Office – jsou důležitým prvkem při komunikaci s klientem, systémy Back-Office – nejsou pro běžného člověka viditelné, neboť přímo nesouvisí s operacemi prováděnými při jednání s klientem. Jsou to interní systémy, které úzce spolupracují s přepážkovými aplikacemi. Zpracovávají data a evidují je, provádějí statistiky a generují příslušné reporty. Podpůrné systémy lze přirovnat ke standardním podnikovým informačním systémům. Ve většině případů mají na starosti chod banky, zpracovávají zaměstnaneckou agendu a poskytují technickou podporu příslušným oddělením.

Mezi podpůrné systémy se „protlačily“ aplikace tzv. “moderních komunikačních kanálů”. Jde o informační systémy pro řízení přímého bankovníctví. Z pohledu klienta poskytují efektivní přístup k účtům a k bankovním produktům prostřednictvím populárního internetového bankovníctví či mobilního bankovníctví (*GSM Banking*). Rovněž zajišťují provoz telefonního centra. Způsob, jak přilákat stávající klienty na nové služby, jak si je udržet nebo jak rozpoznat ty nejvíce profitabilní, vyžaduje vysoce precizní zpracování všech dostupných informací. K tomuto účelu se používají systémy CRM (*Customer Relationship Management*). Ty pracují převážně s daty přepážkových aplikací (Front-Office). Vyhodnocují údaje o tom, co který zákazník

nakupoval, jaké služby využíval či využívá, do jaké věkové a příjmové struktury spadá a mnoho dalších dat. Výsledky vygenerované CRM systémy jsou rozhodujícími při volbě vhodné marketingové strategie pro cílené oslovování klientů s nabídkami nových služeb. Dále napomáhají při rozhodování, kterými nástroji s klientem komunikovat, jak přizpůsobit jednotlivé interní procesy k orientaci na zákazníka, nebo jakým způsobem vyhodnocovat zákazníky a jejich segmenty.

Dokázat identifikovat potenciálního klienta je významným elementem obchodních strategií. Pro tyto účely používají banky různé manažerské informační systémy (MIS), které jsou napojeny na datové sklady (*Datawarehouse*). To jsou samostatné systémy, které v sobě shromažďují informace z dlouhodobého pohledu, tzn. řádově léta až desetiletí. Tím poskytují přístup také k historickým datům, na základě kterých lze provádět kvalitní hloubkové analýzy. Snahou je centralizovat všechny informační systémy na jedno místo a přistupovat k nim ze všech pracovišť pouze přes Internet [INFO 2008].

7.2 Výchozí předpoklady

Analýza dostupných pramenů, popsaných v předchozí teoretické části práce, vedla autora ke zjištění, předpokladům a úvahám uvedeným dále v textu. Tyto předpoklady se opírají o poznatky z dokumentace projektu SQuaRE, zejména konceptu prvků měření kvality uvedeném v kapitole 5, a současně o praktické zkušenosti autora z prostředí banky, která je řazena k velkým podnikům.

Základní sada prvků měření kvality (viz tabulka 3) je velmi univerzální, stejně jako mapování těchto prvků měření kvality na charakteristiky a podcharakteristiky jakosti (tabulka 4 a 5). Otázkou však zůstává, zda je tento výčet dostatečný pro posouzení jakosti software. Zde autor spatřuje celou řadu nedostatků. Jako zásadní pro použití tohoto „návodu“ je již zmíněná univerzálnost. Je uspokojivé, že základní sada může být aplikována na jakýkoli softwarový produkt, ale uživatelé neříká nic o důležitosti (rozdílné míře důležitosti) jednotlivých prvků, ani o tom, zda je tento přehled pro daný software používaný v konkrétním prostředí vyhovující, nebo naopak příliš či málo složitý. Rovněž informace, že výčet není kompletní a může se v čase měnit, nenavádí uživatele, jak má se standardem dále pracovat. Navíc tyto standardy, jak bylo uvedeno v kapitole 3.2.8, jsou psány relativně složitým a méně srozumitelným

jazykem, který může být pro uživatele obtížněji čitelný, a z nich získané informace dále správně interpretovatelné.

Náhled na jakost software je v podnicích, firmách a společnostech různý, stejně jako různá je skladba používaných softwarových produktů. To je důvod, proč je nutné upravit sadu prvků měření kvality, které mají různou míru významnosti podle typu SW produktu, a zároveň mapování na charakteristiky a podcharakteristiky jakosti software pro konkrétní podnik a konkrétní softwarový produkt, implementovaný či vyvíjený v daném prostředí. Toto mapování [mapping] bude rozdílné nejen z důvodu charakteru podniku a typu software, ale i z pohledu role hodnotitele.

Dalším předpokladem pro úspěšný návrh metody je výběr expertů, kteří budou ochotni věnovat svůj čas pro zjištění potřebných informací. Ti musí být detailně seznámeni s cílem výzkumu, a současně poučeni o problematice hodnocení jakosti SW produktů.

Expertní odhad je nejčastěji vyjádřen určitým intervalem, který s vysokou pravděpodobností obsahuje veličinu, která by se získala skutečně přesným měřením. Podstata expertních odhadů spočívá v tom, že experti provedou intuitivně logickou analýzu zkoumaného problému a vyjádří své názory – ohodnocení, které problém specifikují. Vychází se z předpokladu, že skupinový názor expertů v dobře podmíněných úlohách a při dobře volené skupině bývá velmi blízký správnému řešení. Kvalita expertních odhadů závisí především na kvalitě expertů, kteří se odhadování zúčastní. Samozřejmě také závisí na správném postupu vyhodnocení odpovědí expertů. Avšak výběr vhodných expertů je klíčovým faktorem úspěchu. Experta lze definovat takto: expertem rozumíme osobu, která má příslušné znalosti a zkušenosti, jež může prokázat, a které jsou u této osoby všeobecně uznávány. Expertním odhadem rozumíme přijatelný interval hodnot, který s dosti vysokou (často určenou) pravděpodobností obsahuje skutečnou hodnotu, kterou nemůžeme získat bezprostředně měřením. [Lacko 2010].

Autorem byli vybráni a pro účely této práce osloveni tito tři odborníci: ICT Developer, Business architekt, a Aplikační manažer. Přiřazení rolí je následující:

- Vývojář – role vývojáře byla logicky přiřazena k ICT Developerovi, který na základě zadání a požadavků business vyvíjí nebo upravuje používané softwarové produkty.

- Zákazník – role zákazníka byla přiřazena Business architektovi, který staví obchodní procesy a je zodpovědný za business části projektů a formulaci požadavků na vývoj. Rovněž testuje a akceptuje ICT dodávky a může posoudit chování SW produktů z uživatelského hlediska.
- Nezávislý hodnotitel – role nezávislého hodnotitele byla přiřazena Aplikačnímu manažerovi, který dohlíží na používané aplikace a systémy, a zároveň není ovlivněn perspektivou běžného uživatele.

Výběr expertů a následné přiřazení rolí vychází z různých pohledů na jakost popsaných v [Vaníček 2004a]. Důvodem je, že pro různá prostředí a různé softwarové produkty by byl rovněž výběr expertů různý.

V souvislosti s rolemi, respektive s výběrem konkrétních expertů, je velmi důležité zadání managementu – úhel pohledu na jakost informačního systému. Jinými slovy, vymezení přístupu a pohledu na jakost popsané v kapitole 2.3 práce. Dle toho pak budou vybírání experti pro sběr požadavků.

Aby bylo možné výsledky dotazování analyzovat, je třeba zvolit výchozí základnu – tou je v tomto případě informační systém pro řízení klientských dat. Podobný podpůrný informační systém je implementován v každé společnosti tohoto typu. Jde o systém využívaný napříč společnostmi klientskými pracovníky a řadou dalších uživatelů k různým účelům (např. reporting). Prostřednictvím této aplikace se ukládají, spravují a evidují informace o klientech do centrální databáze. Na centrální databázi jsou napojeny další transakční systémy, které využívají informace o klientech z databáze. Klientská data jsou tak vkládána pouze jednou a udržována na jednom místě. Z pochopitelných důvodů nelze uvádět větší detail o tomto systému.

Výše uvedená fakta a poznatky vedly autora k formulaci těchto předpokladů, jak již bylo zmíněno v úvodu této práce:

- Základní sada prvků měření kvality přes svoji univerzálnost nepokrývá praktické potřeby pro hodnocení jakosti, a proto je nutné seznam upravit a rozšířit o další měřitelné prvky.
- Podle konkrétních potřeb hodnotitelů pro daný softwarový produkt je nutné změnit a přepracovat mapování QME na charakteristiky a podcharakteristiky jakosti software.

V následující kapitole je navržena a popsána metoda, která bude sloužit jako obecnější rámec pro použití základního konceptu QME. Předpokladem pro formalizaci metody návrhu QME je ověření tohoto postupu v reálném světě. Provedení tohoto kroku je popsáno v kapitole 7.4.

7.3 Návrh metody a měř pro kvantifikovaný přístup k řízení jakosti

V této kapitole autor prezentuje vlastní návrh metody pro kvantifikovaný přístup k hodnocení jakosti IS. Pro realizaci a použití metody je důležitá spolupráce a alokace čtyř rolí, a to:

- **Management**, který zadává požadavek na hodnocení SW produktu, alokuje zdroje pro provedení hodnocení, nominuje a pověřuje auditora a metodika.
- **Expert**, který je dotazován, jehož expertní odhady slouží jako vstupní data pro vytvoření výsledného modelu.
- **Metodik**, který aplikuje v praxi uvedenou metodu a její výsledky předkládá managementu. Role metodika je dnes ve firmách běžně zastoupena pozicí procesního nebo projektového manažera.
- **Auditor**, který má pověření managementu, a který na základě výsledného modelu provede po aplikaci metody vlastní audit IS.

Metoda je strukturována do čtyř bloků, jak je uvedeno na obrázku 24:

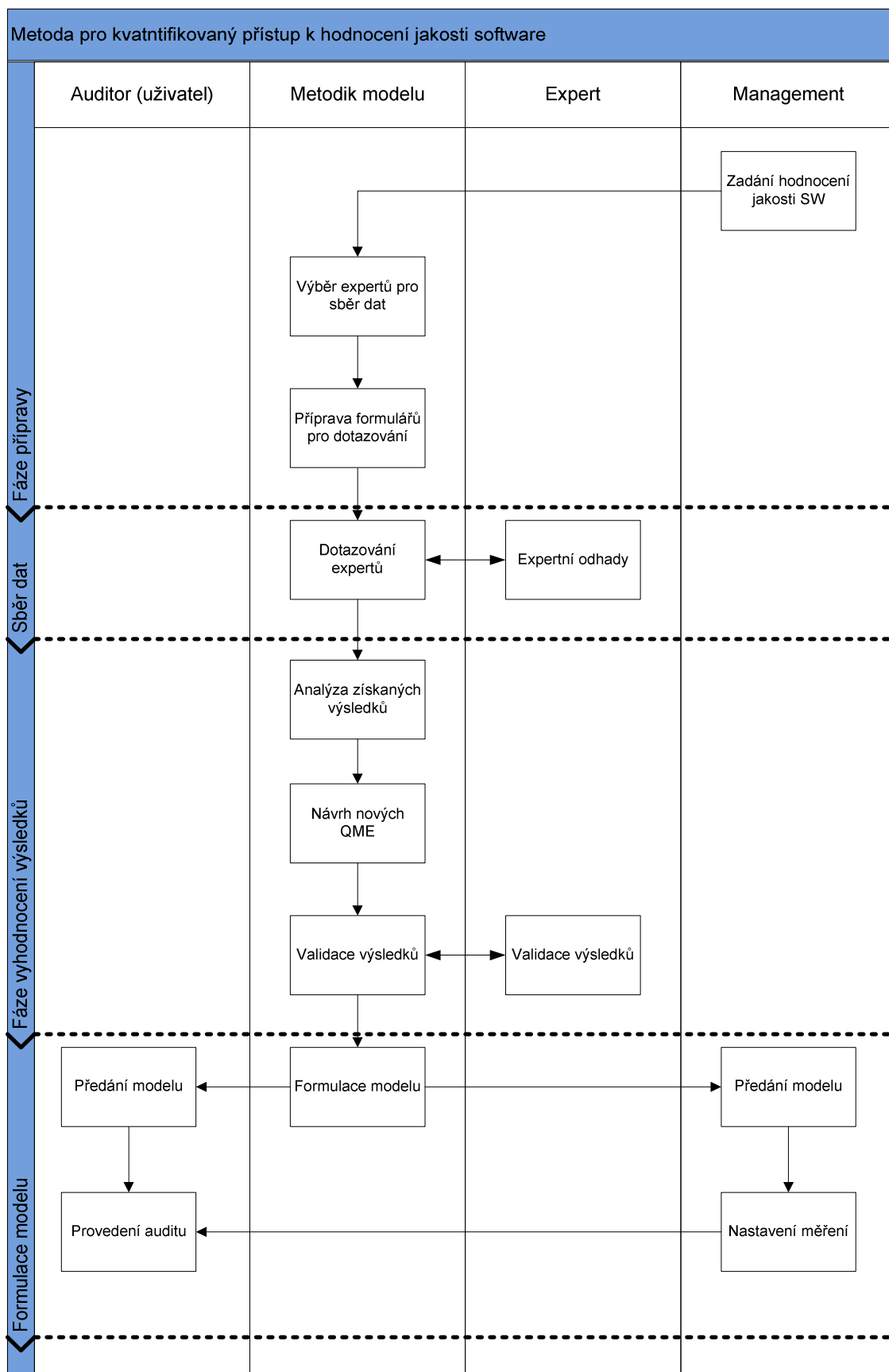
- **Blok přípravy**, který zahrnuje zadání hodnocení IS managementem, výběr hodnoceného software, výběr odborníků pro výzkum a přípravu potřebné dokumentace.
- **Blok sběru dat**, který zahrnuje vlastní výzkum, dotazování vybraných odborníků, sběr požadavků na jakost a zaznamenání dat do příslušných formulářů.
- **Blok vyhodnocení výzkumu**, který zahrnuje analýzu získaných výsledků, návrh nových QME a úpravu mapování na charakteristiky a podcharakteristiky jakosti.
- **Blok formulace modelu**, který zahrnuje vytvoření vlastního modelu na základě analýzy výsledků dotazování, předání modelu managementu a auditorovi, který bude pověřen provedením auditu IS.

Bloky se skládají z konkrétních kroků navržené metody, které jsou jednotlivě popsány a charakterizovány takto:

1. **Zadání hodnocení jakosti IS:** po jednáních s managementem (případně s vedením ICT) by mělo být zřejmé zadání požadavku hodnocení jakosti konkrétního vybraného informačního systému či aplikace ze strany společnosti. Rovněž musí být alokovány zdroje (kapacita pracovníků pro hodnocení, většinou se užívá jednotky pracovního dne MD „*man-day*“). Klíčové je, z jakého pohledu (viz kap. 2.3) na jakost software bude informační systém hodnocen.
2. **Výběr expertů pro sběr dat:** po zadání požadavku na hodnocení jakosti SW je třeba zvolit metodikem odborníky – nejlépe takové, kteří implementovaný SW znají a mohou tak odhadnout a sdělit potřebné informace pro návrh QME. Pokud vybraní odborníci nejsou dostatečně poučeni o problematice hodnocení jakosti SW produktů, autor doporučuje připojit k formulářům pro dotazování stručné vysvětlivky jak základní sady QME, tak i vysvětlivky charakteristik a podcharakteristik jakosti software.
3. **Příprava formulářů pro dotazování:** dokumentace pro samotný výzkum a sběr dat musí být připravena srozumitelnou formou, aby se předešlo případným nedorozuměním z chybné interpretace, a zároveň aby vybraní experti studiem těchto dokumentů netrávili více času, než je nezbytně nutné. Formuláře obsahují kromě prázdného mapování základní sady QME na charakteristiky a podcharakteristiky jakosti prostor pro expertní odhady. Jako základ pro tyto formuláře slouží tabulka ze standardu [ISO/IEC 2011a] uvedená v kapitole 6.2 této práce, tabulka 4 a 5.
4. **Dotazování expertů a vyplnění připravených formulářů:** tento krok zahrnuje a pokrývá samotný výzkum, sběr požadavků na jakost – vysvětlení problematiky, představení formulářů a konkrétních aktivit, které jsou po těchto pracovnících požadovány. Data sebraná dotazováním je v případě nejasnosti nutné s expertem opakovaně ověřit. Ve struktuře formulářů pro sběr dat je připraven prázdný mapping základní sady QME. Vybraný expert vytvoří nové mapování s tím, že nemusí nutně využít všechny QME uvedené v tabulce. Dále navrhne případná další měřitelná kritéria (prvky) pro posouzení jakosti, a rozšíří

tak mapování prvků na charakteristiky a podcharakteristiky jakosti. Příklad takového dokumentu je uveden v kapitole 7.4 práce, tabulka 7, 8.

5. **Analýza získaných výsledků:** hodnotitel provede formální kontrolu úplnosti obdržených dokumentů, které následně posoudí a vyhodnotí. Předpokládá se, že jak nová mapování, tak i návrhy pro další kriteria měření se budou dle jednotlivého pohledu experta lišit. Způsobu vyhodnocení výsledků je věnována kapitola 7.5 práce.
6. **Návrh nových QME:** podle získaných výsledků průzkumu a následného vyhodnocení je třeba navrhnout a popsat včetně všech náležitostí, uvedených v kapitole 6.2 práce tabulka 2, nové prvky měření kvality. Nově přidané – vytvořené prvky měření kvality se začlení do mapování na charakteristiky a podcharakteristiky jakosti a rozšíří tak původní přehled.
7. **Validace výsledků:** navržené a popsané QME je nutné validovat s experty, zejména z důvodu ověření správného pochopení smyslu navrženého prvku a způsobu jeho měření i interpretace metodikem.
8. **Formulace modelu:** na základě vyhodnocení získaných výsledků metodik navrhne výsledný model, který je výstupem této metody. Model obsahuje soubor prvků měření kvality, které budou použity pro konkrétní případ hodnocení IS.
9. **Provedení auditu:** Výsledný model metodik předá jak managementu, tak auditorovi, který je následně po schválení modelu pověřen managementem k provedení auditu vybraného IS.



Obrázek 24: Struktura metody pro kvantifikovaný přístup k jakosti IS³. Zdroj: autor.

³ Pro vytvoření obrázku byl použit MS Visio, který je běžně používanou komerční aplikací.

Pokud dojde k nedorozumění mezi aktéry, nebo k chybné interpretaci jak zadání, tak výsledku, či data pro vyhodnocení nebudou úplná, je nutné konkrétní krok(y) opakovat. Výsledný model z výše uvedené metody návrhu QME je možné použít jako podklad pro komplexní hodnocení či audit informačního systému.

7.4 Ověření metody v praxi

Jedním z výchozích předpokladů návrhu metody, jak již bylo uvedeno, je ověření těchto předpokladů a realizovatelnost metody v praxi. V této kapitole je popsáno konkrétní použití výše uvedené metody.

Na tomto místě je třeba uvést základní (výchozí) fakta nezbytná pro použití navržené metody a zároveň vymežující výchozí předpoklady pro ověření. Jsou uvedena v následující tabulce 6:

Prostředí, ve kterém dochází k ověření navrhované metody	Banka působící na českém trhu
Typ informačního systému implementovaného v prostředí	Podpůrný informační systém pro řízení klientských dat
Fáze životního cyklu zvoleného informačního systému	Provoz
Vybraní experti pro dotazování	ICT Developer, Business Architekt, Aplikační manager

Tabulka 6: Základní fakta pro ověření navrhované metody. Zdroj: Autor.

Prostředí banky působící na českém trhu bylo vybráno z důvodu praxe autora a jeho pracovní pozice a zkušeností, stejně jako informační systém. Jde o podpůrný informační systém pro řízení klientských dat, který prakticky denně využívají řádově stovky pracovníků napříč bankou, a lze jej tedy považovat za dostatečně známý i pro vybrané experty. Těmto kolegům byl autorem vysvětlen cíl dotazování i vlastní metoda návrhu QME. Dále jsou popsány obdržené výsledky od jednotlivých expertů, které byly získány aplikováním navržené metody (dodržením vymezeného postupu):

1. Výsledky Aplikační manažer.

V tabulkách 7 a 8 je uvedeno mapování dle požadavků a expertních odhadů Aplikačního manažera. K tabulkám jsou přiřazeny nové požadavky – prvky měření kvality, které Aplikační manažer považuje pro hodnocení informačního systému v daném prostředí důležité.

Charakteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost						Účinnost		
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
1-Počet přístupů k pomocným funkcím											X			
2-Počet uživatelských problémů		X	X		X	X	X	X	X					
3-Počet záznamů											X	X		X
4-Doba trvání				X	X			X		X		X	X	X
5-Úsilí (v jednotce času)							X	X			X	X	X	
6-Počet poruch systému		X						X	X	X				
7-Počet poruch		X							X	X				
8-Počet závad		X							X	X				
9-Funkční velikost produktu	X		X	X	X	X		X	X				X	
10-Počet přerušení	X							X			X			
11-Počet datových položek											X		X	X
12-Počet chybových hlášení		X						X						
13-Počet chyb		X							X					
14-Počet zpráv											X			
15-Počet kroků (procedur)			X	X				X						
16-Počet úkolů	X		X			X		X	X					
17-Počet testovacích případů								X	X		X		X	
18-Počet use cases	X		X			X	X	X	X	X	X			X
19-Počet operací	X		X			X	X	X	X			X		

Charakteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost						Účinnost		
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
20-Počet kritických chyb		X								X				
21-Velikost databáze														X
22-Velikost paměti														X
Soulad dokumentace a software							X				X			
Předpoklad shody	X	X	X	X	X							X	X	X
Obnovitelnost systému								X	X					
Nedostupnost systému														
Modifikace kódu		X						X					X	

Tabulka 7: mapování mezi QME a jejich použitím pro Funkční přiměřenost, Kompatibilitu, Použitelnost a Účinnost dle Aplikačního manažera. Zdroj: Autor.

Charakteristiky jakosti	Bezporuchovost				Bezpečnost					Udržovatelnost				Přenositelnost			
	Zralost	Dostupnost	Odpornost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
1-Počet přístupů k pomocným funkcím		X		X		X				X	X	X	X	X	X	X	X
2-Počet uživatelských problémů	X	X	X	X	X	X	X	X	X			X		X			X
3-Počet záznamů				X	X	X	X	X	X			X	X		X		
4-Doba trvání		X										X	X	X			
5-Úsilí (v jednotce času)												X	X	X	X		X

Charakteristiky jakosti	Bezporuchovost				Bezpečnost					Udržovatelnost					Přenositelnost		
	Zralost	Dostupnost	Odolnost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
6-Počet poruch systému	X	X	X	X						X	X	X	X	X			
7-Počet poruch	X	X	X	X						X	X	X	X	X			
8-Počet závad	X	X	X	X						X	X	X	X	X			
9-Funkční velikost produktu	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
10-Počet přerušení				X													
11-Počet datových položek				X	X	X	X	X	X			X		X	X		
12-Počet chybových hlášení		X	X									X		X			
13-Počet chyb	X	X	X	X								X		X			
14-Počet zpráv					X		X				X		X		X		
15-Počet kroků (procedur)		X	X		X	X	X										
16-Počet úkolů								X	X						X		
17-Počet testovacích případů														X			
18-Počet use cases				X	X	X	X	X	X	X	X	X	X	X	X	X	X
19-Počet operací			X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
20-Počet kritických chyb	X	X	X	X								X		X			
21-Velikost databáze		X	X	X	X	X	X								X		
22-Velikost paměti		X	X	X				X									
Soulad dokumentace a software								X				X					
Předpoklad shody	X	X	X	X													
Obnovitelnost systému				X							X			X		X	X
Nedostupnost systému	X	X	X	X													X

Charakteristiky jakosti	Bezporuchovost				Bezpečnost				Udržovatelnost				Přenositelnost				
	Zralost	Dostupnost	Odolnost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
Modifikace kódu				X						X	X	X	X	X	X	X	

Tabulka 8: mapování mezi QME a jejich použitím pro Bezporuchovost, Bezpečnost, Udržovatelnost a Přenositelnost dle Aplikačního manažera. Zdroj: Autor.

Aplikačním manažerem přidané požadavky pro hodnocení informačního systému jsou:

- Soulad mezi dokumentací informačního systému a výsledným softwarovým produktem.
- Porovnání očekávaného výsledku informačního systému s předpokládaným výsledkem.
- Obnovitelnost systému jako samostatný prvek měření kvality.
- Celková doba nedostupnosti systému za sledované období.
- Možnost modifikace kódu informačního systému.

Podle těchto doplněných požadavků autor navrhuje tyto nové prvky měření kvality, jak ukazuje následující přehled. Detailnější charakteristiky jednotlivých prvků jsou uvedeny v Příloze A této práce a struktura těchto popisů vychází ze standardu [ISO/IEC 2011a].

Název QME	Popis QME	Způsob měření	Interpretace měřené hodnoty
Soulad dokumentace a software	Zjistit, zda obsah dokumentace k informačnímu systému je konzistentní k implementovanému produktu	$X = A/B$ A = počet bloků informačního systému, které jsou úplně a přesně popsány B = počet bloků informačního systému	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
Předpoklad shody	Porovnat očekávané výstupy systému s aktuálními výstupy z informačního systému	$X = A/t$ A = počet případů, které přesáhly rozdíl mezi aktuálním a očekávaným výsledkem t = čas běhu systému	$0 < X$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 0, tím lépe
Obnovitelnost systému	Zjistit schopnost obnovy systému po abnormální události	$X = A/B$ A = počet případů úspěšné obnovy systému B = počet případů sledovaných abnormálních	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe

		událostí	
Nedostupnost systému	Zjistit průměrnou dobu nedostupnosti informačního systému, pokud dojde k jeho selhání, do doby jeho postupného spuštění	$X = t/A$ $t =$ celkový čas nedostupnosti systému $A =$ počet zaznamenaných zhroucení systému	$0 < X$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 0, tím lépe
Modifikace kódu	Zjistit, zda funkcionality systému mohou být upraveny pomocí modifikace kódu během údržby systému	$X = A/B$ $A =$ počet funkcí informačního systému, které mohou být udržovány úpravou kódu $B =$ počet funkcí informačního systému, které mají být upraveny	$0 < X < 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe

Tabulka 9: Navržené QME dle požadavků Aplikačního manažera. Zdroj: Autor.

2. Výsledky ICT Developer

V tabulkách 10 a 11 je analogicky uvedeno mapování dle požadavků a expertních odhadů ICT Developera. Proti předchozímu výsledku je mapování odlišné, také počet nově přidávaných prvků je dvojnásobný, což ukazuje na nižší míru spokojenosti se základní sadou.

Characteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost						Účinnost		
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
1-Počet přístupů k pomocným funkcím			X											
2-Počet uživatelských problémů	X	X	X		X		X	X	X	X	X	X	X	
3-Počet záznamů									X			X	X	
4-Doba trvání		X		X								X	X	
5-Úsilí (v jednotce času)	X					X	X	X	X	X		X		
6-Počet poruch systému		X										X	X	
7-Počet poruch	X	X							X					

Characteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost						Účinnost		
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
8-Počet závad	X	X	X						X					
9-Funkční velikost produktu	X	X	X		X	X	X	X		X	X	X	X	
10-Počet přerušení				X								X	X	
11-Počet datových položek												X	X	
12-Počet chybových hlášení	X	X				X	X	X	X					
13-Počet chyb	X	X	X			X	X		X				X	
14-Počet zpráv		X				X	X	X						
15-Počet kroků (procedur)			X		X						X			
16-Počet úkolů		X			X						X	X	X	X
17-Počet testovacích případů	X	X	X	X	X	X			X		X	X	X	
18-Počet use cases	X	X	X		X	X	X		X		X			
19-Počet operací					X		X	X	X		X	X		
20-Počet kritických chyb				X									X	
21-Velikost databáze												X	X	
22-Velikost paměti														
Počet funkčních požadavků	X	X	X		X	X	X		X					
Počet systémových požadavků			X	X	X					X		X	X	
Počet hesel uživatelské dokumentace	X		X		X		X	X	X		X			
Počet úkolů (úkonů) popsaných v uživ.návodě	X	X			X	X	X	X		X	X			
Počet hesel administrátorské dokumentace												X	X	

Charakteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost						Účinnost		
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
Počet vstupních/import formátů dat				X					X	X				
Počet výstupních/export formátů dat				X					X	X				
Počet doplňků, pluginů	X					X		X		X				
Počet použitých analytických patternů		X										X	X	
Počet použitých vývojářských patternů			X									X	X	

Tabulka 10: mapování mezi QME a jejich použitím pro Funkční přiměřenost, Kompatibilitu, Použitelnost a Účinnost dle ICT Developera. Zdroj: Autor.

Charakteristiky jakosti	Bezporuchovost				Bezpečnost					Udržovatelnost				Přenositelnost			
	Zralost	Dostupnost	Odolnost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
1-Počet přístupů k pomocným funkcím								X		X	X	X	X				
2-Počet uživatelských problémů	X	X	X	X		X	X	X									
3-Počet záznamů				X		X		X						X		X	X
4-Doba trvání		X				X	X										
5-Úsilí (v jednotce času)		X	X				X								X		
6-Počet poruch systému	X	X	X	X	X	X	X	X	X								
7-Počet poruch	X	X	X	X		X	X	X	X			X		X			

Charakteristiky jakosti	Bezporuchovost				Bezpečnost				Udržovatelnost					Přenositelnost			
	Zralost	Dostupnost	Účinnost vuci vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýty	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
8-Počet závad	X	X	X	X		X	X	X	X			X		X			
9-Funkční velikost produktu	X		X				X		X	X	X	X	X	X	X	X	X
10-Počet přerušení		X	X	X		X	X		X				X				
11-Počet datových položek				X		X		X					X	X	X	X	
12-Počet chybových hlášení	X		X				X						X				
13-Počet chyb	X		X										X				
14-Počet zpráv	X		X														
15-Počet kroků (procedur)									X	X		X	X	X			X
16-Počet úkolů			X			X	X		X	X	X	X					
17-Počet testovacích případů	X		X						X	X	X	X	X	X			X
18-Počet use cases									X	X	X	X	X				
19-Počet operací		X		X		X		X	X								
20-Počet kritických chyb			X	X	X	X	X	X	X								
21-Velikost databáze				X		X		X					X	X	X	X	
22-Velikost paměti																	
Počet funkčních požadavků	X								X	X	X	X	X				
Počet systémových požadavků	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X
Počet hesel uživatelské dokumentace	X									X							
Počet úkolů (úkonů) popsaných v uživ.nápravě	X										X	X					
Počet hesel administrátorské dokumentace			X	X			X	X		X		X		X	X		

Charakteristiky jakosti	Bezporuchovost				Bezpečnost				Udržovatelnost				Přenositelnost				
	Zralost	Dostupnost	Odolnost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analyzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
Počet vstupních/import formátů dat				X		X	X							X	X		X
Počet výstupních/export formátů dat				X		X	X							X	X		X
Počet doplňků, pluginů										X	X		X		X	X	
Počet použitých analytických patternů	X							X		X	X	X	X		X		X
Počet použitých vývojářských patternů	X		X			X		X		X	X	X	X		X		X

Tabulka 11: mapování mezi QME a jejich použitím pro Bezporuchovost, Bezpečnost, Udržovatelnost a Přenositelnost dle ICT Developera. Zdroj: Autor.

ICT Developerem přidané požadavky pro hodnocení informačního systému jsou:

- Implementované funkční požadavky na informační systém dle funkční specifikace.
- Implementované systémové požadavky dle zadané specifikace.
- Úroveň detailu, přehlednosti a srozumitelnosti uživatelské dokumentace.
- Úroveň detailu, přehlednosti a srozumitelnosti uživatelské nápovědy.
- Úroveň detailu, přehlednosti a srozumitelnosti administrátorské dokumentace.
- Implementovaný rozsah požadovaných formátů dat pro import.
- Implementovaný rozsah formátů dat pro export.
- Implementovaný rozsah požadovaných doplňků systému.
- Míra použitých analytických best-practice postupů.
- Míra použitých vývojářských best-practice postupů.

V následujícím přehledu jsou uvedeny nové prvky měření kvality podle požadavků ICT Developera. Detailnější charakteristiky jednotlivých prvků jsou uvedeny v Příloze A této práce a struktura těchto popisů vychází ze standardu [ISO/IEC 2011a].

Název QME	Popis QME	Způsob měření	Interpretace měřené hodnoty
Počet funkčních požadavků	Zjistit, do jaké míry jsou realizovány specifikované funkční požadavky na informační systém.	$X = A/B$ A = počet realizovaných funkčních požadavků B = počet požadovaných funkčních požadavků	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
Počet systémových požadavků	Zjistit, do jaké míry jsou realizovány specifikované systémové požadavky na informační systém.	$X = A/B$ A = počet realizovaných systémových požadavků B = počet požadovaných systémových požadavků	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
Počet hesel uživatelské dokumentace	Zjistit, do jaké míry je detailně a srozumitelně vytvořena uživatelská dokumentace pro pochopení uživatelem.	$X = A/B$ A = počet aktivit, které lze správně provést po přečtení dokumentace B = počet aktivit uživatelské dokumentace	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
Počet úkolů (úkonů) popsaných v uživ. nápovědě	Zjistit, jaká část úkonů nebo typů úkonů je správně pochopena po přečtení dokumentace.	$X = A/B$ A = počet úkonů, které lze provést po přečtení nápovědy B = počet všech úkonů v uživatelské nápovědě	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
Počet hesel administrátorské dokumentace	Zjistit, do jaké míry je detailně vytvořena administrátorská dokumentace	$X = A/B$ A = počet aktivit, které lze správně provést po přečtení dokumentace B = počet aktivit administrátorské dokumentace	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
Počet vstupních/import formátů dat	Porovnat implementované formáty vstupních dat s požadavky na import dat.	$X = (A/B) \cdot 100$ A = realizovaný počet vstupních formátů dat B = požadovaný počet vstupních formátů dat	$0 \leq X \leq 100$ Procenta, čím blíže je poměrové číslo k 100, tím lépe
Počet výstupních/export formátů dat	Porovnat implementované formáty výstupních dat s požadavky na export dat.	$X = (A/B) \cdot 100$ A = realizovaný počet výstupních formátů dat B = požadovaný počet výstupních formátů dat	$0 \leq X \leq 100$ Procenta, čím blíže je poměrové číslo k 100, tím lépe
Počet doplňků, pluginů	Zjistit, do jaké míry jsou realizovány specifikované doplňky na informační systém.	$X = A/B$ A = počet realizovaných doplňků B = počet požadovaných doplňků systému	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
Počet použitých analytických patternů	Zjistit, do jaké míry byly použity standardní analytické best-practice postupy.	$X = (A/B) \cdot 100$ A = počet použitých analytických patternů B = počet použitých postupů	$0 \leq X \leq 100$ Procenta, čím blíže je poměrové číslo k 100, tím lépe

Počet použitých vývojářských paternů	Zjistit, do jaké míry byly použity standardní vývojářské best-practice postupy.	$X = (A/B) \cdot 100$ A = počet použitých vývojářských paternů B = počet použitých postupů	$0 \leq X \leq 100$ Procenta, čím blíže je poměrové číslo k 100, tím lépe
---	---	--	--

Tabulka 12: Navržené QME dle požadavků ICT Developera. Zdroj: Autor.

3. Výsledky Business architekt

V tabulkách 13 a 14 je uvedeno mapování dle požadavků a expertních odhadů Business architekta. Proti výše uvedeným výsledkům, jak Aplikačního manažera, tak ICT Developera, je mapování odlišné, avšak zásadní rozdíl je v tom, že zde nejsou nově přidané prvky měření kvality. Sada těchto prvků je tedy vyhovující, nespokojenost je patrná s jejich mapováním na charakteristiky a podcharakteristiky jakosti.

Characteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost					Účinnost			
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
1-Počet přístupů k pomocným funkcím						X		X			X			
2-Počet uživatelských problémů	X	X				X	X	X	X	X	X			
3-Počet záznamů											X			X
4-Doba trvání				X	X			X			X	X		
5-Úsilí (v jednotce času)		X					X	X					X	
6-Počet poruch systému														
7-Počet poruch		X							X					
8-Počet závad									X					
9-Funkční velikost produktu	X												X	
10-Počet přerušení	X													
11-Počet datových položek										X				

Charakteristiky jakosti	Funkční přiměřenost			Kompatibilita		Použitelnost					Účinnost			
	Funkční úplnost	Funkční korektnost	Funkční přiměřenost	Koexistence	Interoperabilita	Vhodnost účelu	Naučitelnost	Snadnost obsluhy	Ochrana proti chybám uživatele	Atraktivnost	Přístupnost	Časové chování	Nároky na zdroje	Kapacita
QME/ subcharakteristiky														
12-Počet chybových hlášení						X					X			X
13-Počet chyb		X							X					
14-Počet zpráv						X					X			
15-Počet kroků (procedur)			X					X						
16-Počet úkolů	X		X			X			X					
17-Počet testovacích případů	X							X	X		X			
18-Počet use cases	X						X	X	X	X	X			
19-Počet operací	X						X	X	X	X	X			
20-Počet kritických chyb		X								X				
21-Velikost databáze												X		X
22-Velikost paměti														X

Tabulka 13: mapování mezi QME a jejich použitím pro Funkční přiměřenost, Kompatibilitu, Použitelnost a Účinnost dle Business architekta. Zdroj: Autor.

Charakteristiky jakosti	Bezporuchovost				Bezpečnost					Udržovatelnost				Přenositelnost			
	Zralost	Dostupnost	Odolnost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
1-Počet přístupů k pomocným funkcím																	
2-Počet uživatelských problémů	X	X	X	X	X	X	X	X	X						X	X	X

Charakteristiky jakosti	Bezporuchovost				Bezpečnost					Udržovatelnost					Přenositelnost		
	Zralost	Dostupnost	Odolnost vůči vadám	Obnovitelnost	Důvěrnost	Integrita	Nenarušení	Dohledatelnost	Autenticita	Modularita	Znovupoužitelnost	Možnost analýzy	Modifikovatelnost	Testovatelnost	Možnost adaptace	Instalovatelnost	Nahraditelnost
QME/ subcharakteristiky																	
3-Počet záznamů				X	X	X	X	X	X						X		
4-Doba trvání												X	X	X			
5-Úsilí (v jednotce času)												X	X	X			
6-Počet poruch systému												X	X	X			
7-Počet poruch	X	X	X	X						X	X	X	X	X			
8-Počet závad	X	X	X	X						X	X	X	X	X			
9-Funkční velikost produktu	X	X	X	X													
10-Počet přerušení														X			
11-Počet datových položek				X	X	X	X	X	X						X		
12-Počet chybových hlášení		X	X												X		
13-Počet chyb	X	X	X	X													
14-Počet zpráv																	
15-Počet kroků (procedur)																	
16-Počet úkolů															X		
17-Počet testovacích případů														X			
18-Počet use cases			X	X	X	X	X	X	X								
19-Počet operací			X	X	X	X	X	X	X								
20-Počet kritických chyb	X	X	X	X										X			
21-Velikost databáze																	
22-Velikost paměti																	

Tabulka 14: mapování mezi QME a jejich použitím pro Bezporuchovost, Bezpečnost, Udržovatelnost a Přenositelnost dle Business architekta. Zdroj: Autor.

7.5 Vyhodnocení výsledků

Aby autorem navržená metoda kvantifikovaného přístupu k hodnocení jakosti IS našla své uplatnění pro praktické používání, je třeba definovat výstup, který bude po aplikaci metody obdrženým výsledkem. Výsledkem užití navržené metody je **model** prvků měření kvality, který je relevantní pro konkrétní softwarový produkt používaný v daném prostředí. Tento model je vlastně optimalizací odhadů expertů, jejich požadavků na hodnocení jakosti, podle kterých byly navrženy prvky měření kvality, jak je popsáno v předchozí kapitole práce.

Pro vyhodnocení výsledků obdržených metodou dotazování vybraných expertů je třeba, aby byla jednotlivým prvkům měření kvality přiřazena priorita. Tento stupeň důležitosti prvku měření kvality je podle odhadu expertů uveden v následujících tabulkách. Autor opět zvolil metodu dotazování pro přiřazení těchto priorit. Priorita 3 je nejvyšší, Priorita 1 znamená nejnižší důležitost prvku v přehledu.

Název QME	Přiřazení priorit - Aplikační manažer		
	Priorita 3	Priorita 2	Priorita 1
1-Počet přístupů k pomocným funkcím			X
2-Počet uživatelských problémů	X		
3-Počet záznamů		X	
4-Doba trvání		X	
5-Úsilí (v jednotce času)		X	
6-Počet poruch systému	X		
7-Počet poruch	X		
8-Počet závad	X		
9-Funkční velikost produktu		X	
10-Počet přerušení	X		
11-Počet datových položek		X	
12-Počet chybových hlášení			X
13-Počet chyb	X		
14-Počet zpráv		X	
15-Počet kroků (procedur)		X	
16-Počet úkolů		X	
17-Počet testovacích případů			X
18-Počet use cases	X		
19-Počet operací	X		
20-Počet kritických chyb	X		
21-Velikost databáze		X	
22-Velikost paměti		X	
Soulad dokumentace a software	X		
Předpoklad shody		X	
Obnovitelnost systému	X		
Nedostupnost systému	X		
Modifikace kódu		X	

Tabulka 15: Přiřazení priorit QME podle odhadu Aplikačního manažera. Zdroj: Autor.

Název QME	Přiřazení priorit - ICT Developer		
	Priorita 3	Priorita 2	Priorita 1
1-Počet přístupů k pomocným funkcím			X
2-Počet uživatelských problémů	X		
3-Počet záznamů		X	
4-Doba trvání		X	
5-Úsilí (v jednotce času)			X
6-Počet poruch systému	X		
7-Počet poruch		X	
8-Počet závad		X	
9-Funkční velikost produktu	X		
10-Počet přerušení		X	
11-Počet datových položek	X		
12-Počet chybových hlášení			X
13-Počet chyb	X		
14-Počet zpráv			X
15-Počet kroků (procedur)		X	
16-Počet úkolů		X	
17-Počet testovacích případů			X
18-Počet use cases		X	
19-Počet operací	X		
20-Počet kritických chyb	X		
21-Velikost databáze		X	
22-Velikost paměti			X
Počet funkčních požadavků	X		
Počet systémových požadavků	X		
Počet hesel uživatelské dokumentace			X
Počet úkolů (úkonů) popsaných v uživ.nápovědě		X	
Počet hesel administrátorské dokumentace		X	
Počet vstupních/import formátů dat			X
Počet výstupních/export formátů dat			X
Počet doplňků,pluginů			X
Počet použitých analytických paternů	X		
Počet použitých vývojářských paternů	X		

Tabulka 16: Přiřazení priorit QME podle odhadu ICT Developera. Zdroj: Autor.

Název QME	Přiřazení priorit - Business architekt		
	Priorita 3	Priorita 2	Priorita 1
1-Počet přístupů k pomocným funkcím			X
2-Počet uživatelských problémů	X		
3-Počet záznamů		X	
4-Doba trvání		X	
5-Úsilí (v jednotce času)			X
6-Počet poruch systému	X		
7-Počet poruch		X	
8-Počet závad		X	
9-Funkční velikost produktu	X		
10-Počet přerušení	X		
11-Počet datových položek			X
12-Počet chybových hlášení		X	
13-Počet chyb		X	
14-Počet zpráv			X

15-Počet kroků (procedur)		X	
16-Počet úkolů		X	
17-Počet testovacích případů	X		
18-Počet use case	X		
19-Počet operací		X	
20-Počet kritických chyb	X		
21-Velikost databáze			X
22-Velikost paměti			X

Tabulka 17: Přiřazení priorit QME podle odhadu Business architekta. Zdroj: Autor.

Aby bylo ohodnocení jednotlivých prvků měření kvality úplné, je třeba ještě ohodnotit – přiřadit priority nově přidaným prvkům ostatními experty – tak, jak je tomu v případě prvků základní sady. Toto ohodnocení ukazuje následující tabulka 18.

Název QME	Přiřazení priorit ICT Developer			Přiřazení priorit Business architekt		
	Priorita 3	Priorita 2	Priorita 1	Priorita 3	Priorita 2	Priorita 1
Soulad dokumentace a software	X			X		
Předpoklad shody		X			X	
Obnovitelnost systému	X				X	
Nedostupnost systému		X		X		
Modifikace kódu		X				X
Název QME	Přiřazení priorit Aplikační manager			Přiřazení priorit Business architekt		
	Priorita 3	Priorita 2	Priorita 1	Priorita 3	Priorita 2	Priorita 1
Počet funkčních požadavků	X			X		
Počet systémových požadavků	X			X		
Počet hesel uživatelské dokumentace		X		X		
Počet úkolů v uživ.návodě			X	X		
Počet hesel admin. dokumentace		X			X	
Počet vstupních/import formátů dat			X		X	
Počet výstupních/export formátů dat			X		X	
Počet doplňků, pluginů		X		X		
Počet použitých analytických paternů		X				X
Počet použitých vývojářských paternů		X				X

Tabulka 18: Přiřazení priorit nově doplněným QME. Zdroj: Autor.

Pokud bychom chtěli pro měření použít všechny navržené prvky měření kvality, je třeba si uvědomit, že by takové hodnocení bylo relativně velmi nákladné. Je to zejména z důvodu časové náročnosti provedení všech těchto měření. V tomto případě jde o dalších 15 nových prvků navržených dle požadavků expertů; požadavků na měření jakosti SW však může být i mnohonásobně více, podle počtu a odhadů dotázaných expertů. Proto je nutné stanovit množinu a její maximální počet prvků pro měření.

Stanovení rozsahu množiny, kvantifikace počtu prvků měření kvality, je pro tento případ autorem odhadnut na 20. Tento údaj vychází z konzultací s odborníky, kteří tento odhad doporučují z důvodu relativně jednoduchého použití a časové náročnosti této metody. Pro stanovení množiny těchto prvků, které budou tvořit výsledný model, bylo použito ohodnocení, resp. přiřazení priorit, jednotlivých QME, jak ukazují tabulky 15 – 18 v této kapitole. Grafické znázornění ohodnocení QME experty ukazuje obrázek 25.

7.5.1 Vytvoření výsledného modelu

Kvantifikace konkrétních prvků měření kvality do výsledného modelu je stanovena kombinací přiřazených priorit jednotlivým prvkům měření kvality experty a shody v odhadech těchto expertů. Pro tento účel, jak už bylo uvedeno, byli autorem vybráni a osloveni tito odborníci: Business architekt, Aplikační manažer, a ICT Developer.

Výběr QME do výsledného modelu pro zvolený aplikační software, informační systém pro řízení klientských dat, je popsán v těchto krocích:

- Jednotlivým prvkům měření kvality jsou přiřazeny priority (váhy) tak, aby byly ohodnoceny všechny prvky všemi dotazovanými experty, tedy i ty, které byly navrženy na základě požadavků vybraných expertů.
- Tato struktura QME a vah byla znázorněna z důvodu větší přehlednosti graficky. Konstrukce 3D grafu je: na ose x jsou všechny prvky měření kvality (QME), na ose y jsou priority (Priorita 3, Priorita 2, Priorita 1), a na ose z vybraní experti (BA – Business architekt, AM – Aplikační manažer, ICT – ICT Developer). Graf pak ukazuje přiřazení priorit ke každému prvku podle ohodnocení expertů, jak ukazuje Obrázek 25.
- Jako první jsou do výstupu (modelu) vybrány ty prvky, kde existuje absolutní shoda expertů v přiřazení nejvyšší priority konkrétnímu QME (např. „Počet poruch systému“: Aplikační manažer – Priorita 3, Business architekt – Priorita 3, ICT Developer – Priorita 3).
- Podle počtu vybraných prvků jsou do modelu dále vybrány prvky do doplnění počtu definované množiny, pokud nebyla naplněna prvky s absolutní shodou přiřazení nejvyšší priority, kde existuje částečná shoda mezi experty v přiřazení nejvyšších priorit QME (např. „Obnovitelnost systému“: Aplikační manažer – Priorita 3, Business architekt – Priorita 2, ICT Developer – Priorita 3).

- Prvky, kterým byly přiřazeny nejnižší priority, nebo u kterých existuje neshoda mezi experty (každý z expertů uvedl jiný odhad ve spektru nejnižší a nízké priority), nebudou do výsledného modelu zařazeny.

Podle tohoto postupu výběru prvků měření kvality je pro informační systém pro řízení klientských dat kvantifikován výsledný model dvaceti prvků, jak ukazuje následující Tabulka 19:

#	Název QME	#	Název QME
1	Počet uživatelských problémů	11	Soulad dokumentace a software
2	Počet poruch systému	12	Obnovitelnost systému
3	Počet poruch	13	Nedostupnost systému
4	Počet závad	14	Počet funkčních požadavků
5	Funkční velikost produktu	15	Počet systémových požadavků
6	Počet přerušení	16	Počet hesel administrátorské dokumentace
7	Počet chyb	17	Počet úkonů popsaných v uživ.návodě
8	Počet use cases	18	Počet doplňků, pluginů
9	Počet operací	19	Počet použitých analytických paternů
10	Počet kritických chyb	20	Počet použitých vývojářských paternů

Tabulka 19: Výsledný model QME. Zdroj: Autor.

Tento výsledný model, který obdrží metodik, pak podle formalizované metody popsané v kapitole 7.3. práce je předán jak managementu, tak auditorovi, kterému bude sloužit jako podklad pro provedení hodnocení jakosti IS. Jeho provedení závisí na zadání managementu společnosti, a často komunikaci managementu a ICT managementu jako podpoře business. Tímto vztahem se zabývá např. publikace [Dohnal, Příklenk 2011]. Jak již bylo uvedeno, výsledný model je optimalizací expertních odhadů, které vycházejí z požadavků na jakost konkrétního software. Jinými slovy, výsledný model je platný pouze pro informační systém charakterizovaný ve výchozích předpokladech. Je tedy zřejmé, že výstupní množina QME bude jiná v závislosti na druhu hodnoceného software, prostředí, kde je IS implementován, dále na expertech a jejich požadavcích, stanoveném počtu QME pro měření, a v neposlední řadě také na managementu, který stanovuje rozsah auditu a alokuje potřebné zdroje pro jeho provedení.

7.5.2 Vytvoření výsledného modelu pomocí software MaxAgr

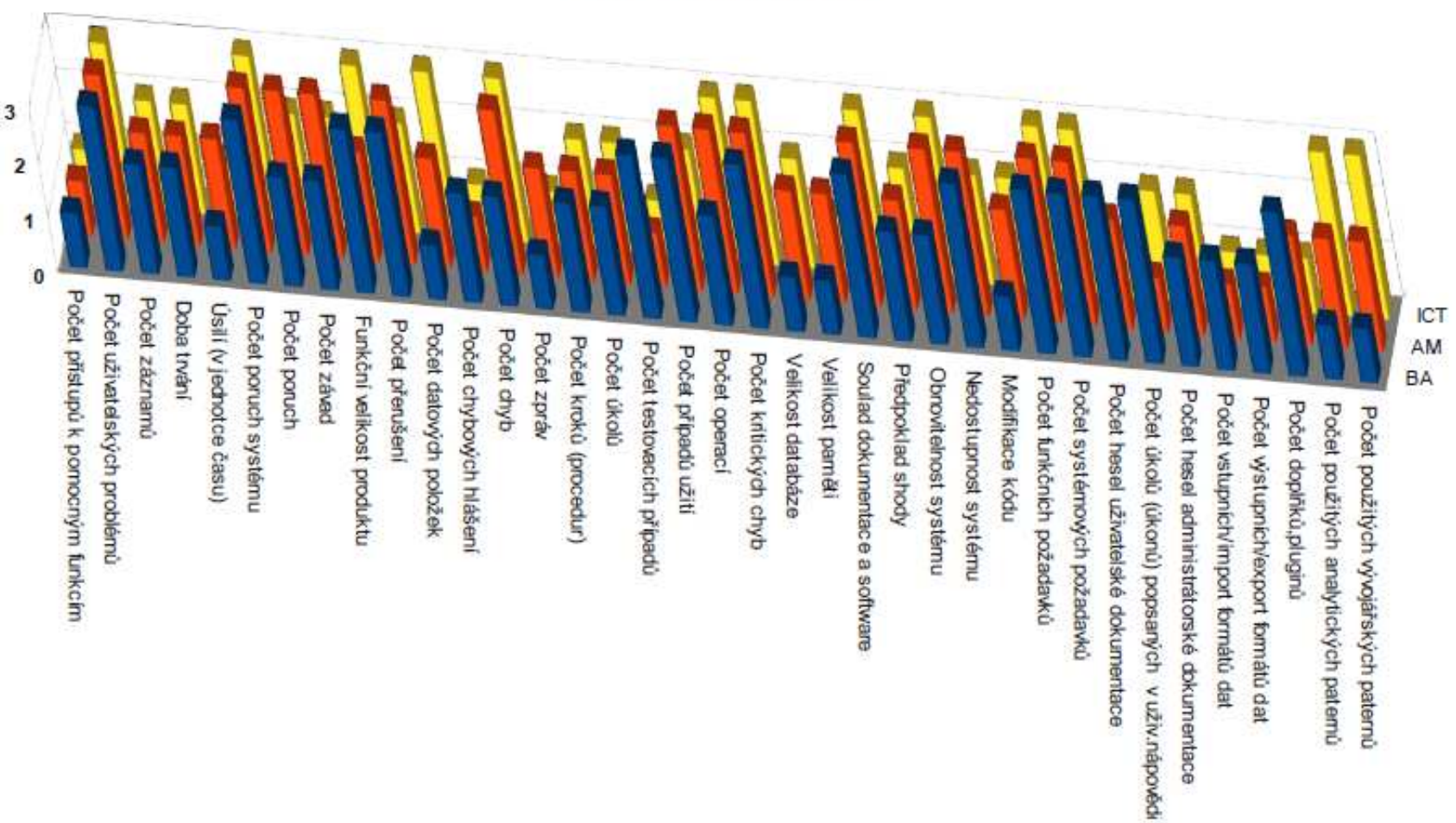
Jako další příklad způsobu vytvoření výsledného modelu lze použít specializovaný software MaxAgr, který byl vyvinut na katedře informačního inženýrství pro optimalizaci dohody vypracované z návrhu expertů. Software MaxAgr je určen pro stanovení hodnoty expertně posuzovaného parametru, která odpovídá nejlepší shodě

názorů posuzujících expertů. Může se jednat o stanovení jak binární, tak číselné hodnoty, a tato hodnota může být vyjádřena jak ostrými, tak i fuzzy čísly nebo fuzzy intervaly. Výpočet optimálního odhadu je založen na algoritmu nově vyvinutého agregačního operátoru Agreement. Software nabízí možnost stanovit odhad i podle vybraných klasických agregačních operátorů a umožňuje tak lepší interpretaci výsledku expertního posouzení při skupinovém rozhodování za podmínek neurčitosti [Vrana et al. 2012].

Pro vytvoření modelu prvků měření kvality byly použity, stejně jako v postupu popsáném v předcházející kapitole, priority přiřazené experty jednotlivým prvkům. Tedy vstupem pro výpočet maximální shody expertních odhadů pomocí programu MaxAgr jsou přiřazené priority, na základě kterých se, vždy pro každý prvek zvlášť, vypočte a zobrazí míra maximální shody. Posouzením obdržených hodnot lze konstatovat, že použitím software MaxAgr bylo dospěno k stejnému výslednému modelu, který je prezentován výše. Důvodem pro tuto skutečnost je pravděpodobně nižší počet vybraných a dotázaných expertů, a tím i nižší množina vah pro výpočet maximální shody.

Použití software MaxAgr předpokládá větší množinu expertů a expertních odhadů, než je obsahem této práce. Způsob vyhodnocování výsledků pomocí tohoto programu je velmi efektivní a lze jej tak považovat jako námět pro další výzkum v této oblasti.

Přiřazení priorit QME



Obrázek 25: Přiřazení priorit QME. Zdroj: Autor.

8 Závěr

8.1 Shrnutí disertační práce

Tato disertační práce je věnována kvantifikovanému přístupu k jakosti informačního zabezpečení pro podporu evaluace informačních technologií. V úvodu jsou nastíněny důvody vzniku této práce, stejně jako její struktura, a hlavní i dílčí cíle tohoto zpracování. V úvodní kapitole jsou rovněž uvedeny limity a omezení řešení, a přehled souvisejících tematických celků, které nejsou obsahem disertace. Práce obsahuje rešeršní část – přehled současného stavu problematiky, kterou reprezentují kapitoly 2 až 6, a výsledky disertační práce, které jsou obsahem kapitoly 7.

Hlavním cílem disertační práce bylo navrhnout metodu pro kvantifikovaný přístup k řízení jakosti softwarových produktů. Dílčí cíle disertační práce byly: podle dostupných zdrojů analyzovat a zhodnotit projekt SQuaRE a navrhnout prvky měření kvality na základě sběru uživatelských požadavků na jakost software.

V rešeršní části práce jsou uvedeny poznatky o jakosti softwarových produktů. Vysvětlen je pojem jakosti, jakost produktu a procesu, popsány jsou různé pohledy na jakost IS. Dále je nastíněn přehled současného stavu mezinárodní normalizace kvality software, kde zvláštní pozornost je věnována projektu SQuaRE, který má za cíl nahradit stávající normy. Část práce o jakosti IS uzavírá princip a způsob měření jakosti IS. Jako navazující téma je uveden obecný postup provádění auditu IS, který není součástí navrženého postupu. Lze konstatovat fakt, že problematika jakosti softwarových produktů i auditu informačních systémů je v odborné literatuře popsána dostatečně.

Zvláštní pozornost je věnována konceptu prvků měření kvality (QME), který je relativně novinkou v mezinárodní normalizaci jakosti software. V 6. kapitole práce je kromě konstrukce uvedena základní sada těchto prvků, i jejich mapování na charakteristiky a podcharakteristiky jakosti software. Tyto informace z mezinárodního standardu byly naprosto nezbytné pro navržené řešení práce a pro autora byly jedním z výchozích předpokladů pro návrh metody.

Na základě teoretických poznatků uvedených v rešeršní části práce a vlastních zkušeností autor prezentuje formalizovaný návrh metody a měř pro kvantifikovaný přístup k řízení jakosti software. Nejprve byly autorem vymezeny a formulovány výchozí předpoklady pro návrh metody.

Vlastní metoda, charakterizovaná v kapitole 7.3 práce, je rozdělena do čtyř základních fází: fáze přípravy, sběru dat, vyhodnocení výsledků a formulace modelu. Každý blok metody se skládá z konkrétních kroků, z nichž by žádný neměl být vynechán. Nejprve musí být managementem firmy zadáno hodnocení konkrétního softwarového produktu a alokovány zdroje pro provedení hodnocení. Dále je nutné vybrat experty pro sběr dat, na základě kterých budou po analýze těchto získaných informací navrženy prvky měření kvality. Po jejich validaci s experty je třeba kvantifikovat model prvků měření kvality. Navržená metoda je kombinací teoretické a empirické části disertace.

Ověření metody autor provedl ve finančním sektoru, v prostředí banky. Pro sběr dat byli vybráni tři typově různí experti: Business architekt, Aplikační manažer a ICT Developer. Softwarový produkt představuje informační systém pro řízení klientských dat. Aplikací navržené metody autor získal výsledky v podobě mapování základní sady prvků měření kvality na charakteristiky a podcharakteristiky jakosti, a požadavky na jakost software. Podle nich pak byly navrženy nové prvky měření kvality.

Vyhodnocení výsledků, které je následně uvedeno v kapitole 7.5, je zaměřeno na způsob výběru prvků měření kvality do výsledného modelu pro zvolený softwarový produkt. Výběr těchto prvků je založen na shodě expertních odhadů v přiřazení priorit každému prvku uvedené množiny. Toto ohodnocení je znázorněno graficky a výběr prvků do modelu je dán shodou v přiřazení nejvyšších priorit. Výsledkem (výstupem) navržené metody je model prvků měření kvality, který lze použít pro hodnocení, audit konkrétního informačního systému používaného v konkrétním prostředí společnosti.

8.2 Přínosy disertační práce

Navržená metoda je optimalizací odhadů expertů a jejich požadavků na jakost software. Jde o způsob, jakým rigorózně stanovit potřebné atributy k měření, tj. atributy v modelu. Jinými slovy, tato metoda ukazuje způsob sběru uživatelských požadavků založených na charakteristikách jakosti. Formulované předpoklady byly návrhem metody a jejím ověřením v praxi potvrzeny. Autor se domnívá, že by tak cíle této disertační práce mohly být splněny. Výhody navržené metody jsou:

- Relativně jednoduchá realizace a použití metody bez nutnosti další investice do dostupných komerčních produktů.

- Efektivnost – role potřebné pro provedení metody jsou v podniku již typicky obsaženy.
- Kritéria pro návrh prvků měření kvality jsou sebrána od expertů, kteří mají znalost hodnoceného softwarového produktu.
- Kvantifikovaný přístup – přístup pracuje s číselnými kvantifikacemi a poskytuje tak objektivní aparát, jehož výstupy je možné dále zpracovávat a interpretovat.
- Měření lze opakovat po určité době stejnou metodou, ve stejném prostředí na totožném software. je tak možné sledovat a vyhodnocovat změny a trendy.
- Použití stejné metody pro různé softwarové produkty ve společnosti – jakost lze porovnat, protože je hodnocena stejným způsobem.
- Výsledky použití metody mohou sloužit jako podklad pro vytvoření interního SLA (*Service Level Agreement*).

Navržená metoda přes svoje výhody uvedené viz výše a časovou nenáročnost nemá být konkurencí, nebo dokonce nahrazením profesionálních nástrojů pro řízení podnikové informatiky, jako je např. COBIT [COBIT 2007], nebo Balanced Scorecard [BSC 2010], [Wagner 2009], apod. Tato metoda má své uplatnění všude tam, kde buď nejsou k dispozici profesionální komerční nástroje pro audit IS, nebo nejsou vhodné softwarové produkty pro hodnocení. Nicméně, tato metoda může být použita pro hodnocení kvality prakticky kdykoli ve všech typech podniků a společnostech, které požadují hodnocení kvality software, bez vysokých nákladů na čas a zdroje. Přínosy obsažené v disertační práci lze charakterizovat takto:

- Práce obsahuje analýzu a hodnocení stavu projektu SQuaRE a překlad klíčových konceptů, které vychází z originálních textů norem a český překlad není zatím vytvořen.
- Práce obsahuje návrh prvků měření kvality podle odhadů požadavků expertů na jakost konkrétního softwarového produktu.
- Práce obsahuje v praxi použitelnou metodu pro kvantifikovaný přístup k jakosti software.

Využití metody autor spatřuje spíše v menších firmách a společnostech pro audit informačních systémů. Metoda je efektivní a využívá k aplikaci rolí v podniku obsažených, tedy bez nutnosti nákupu externích zdrojů (např. lidských zdrojů).

8.3 Diskuse a náměty k dalšímu výzkumu

Pro další výzkum lze doporučit aplikaci metody v jiném odvětví, než je bankovníctví. Výsledný model bude pravděpodobně rozdílný od modelu uvedeného v této práci, protože budou rozdílné výchozí předpoklady pro použití metody. Dalším námětem by mohla být analýza a výzkum, zda by bylo možné navrženou metodu využít a včlenit do komerčních nástrojů pro řízení podnikové informatiky.

Evaluace informačního zabezpečení podniku může být prováděna z různých pohledů: zákazníka, vývojáře, a nezávislého hodnotitele. Každá tato skupina má jiná kritéria (požadavky) hodnocení jakosti. Z tohoto důvodu je klíčový cíl evaluace a pohled na jakost softwarového produktu, který má být hodnocen. Proto je velmi důležitá role managementu, který plní roli nejen jako zadavatel provádění těchto hodnocení a alokace zdrojů, ale i zákazníka, který obdrží jeho výsledek a následně podle něj přijímá případná opatření.

Přestože ta část populace, která využívá informační technologie, je zároveň koncovým uživatelem, existuje jistá diverzifikace – to se ukázalo např. při výběru expertů pro dotazování. Ačkoli navržená metoda má široké uplatnění, není určena pro každého uživatele ICT. Dle názoru autora je tento postup efektivní pro ty pracovníky, kteří jsou často např. v projektech „mostem“ mezi business částí a ICT, tedy nejen pro procesní či projektové manažery. Tato role je v praxi zastoupena pozicí IT Business Analytika, nebo obecně Business Architekta, a těchto odborníků trh pracovních pozic rozhodně není přesycen.

Autor se domnívá, že výsledky obsažené v této disertační práci přispívají k rozvoji poznání v problematice hodnocení jakosti softwarových produktů a dávají managementu firem do rukou jednoduchý a účinný nástroj pro provádění takových hodnocení, ze kterých vedení obdrží nejen informace o kvalitě provozovaných informačních technologiích, ale i důležitá data pro strategická rozhodování při plnění svých cílů, a může tak ovlivňovat efektivitu používaného aplikačního software.

Seznam obrázků

OBRÁZEK 1: STRUKTURA DISERTAČNÍ PRÁCE. ZDROJ AUTOR.	12
OBRÁZEK 2: MODEL KVALITY SOFTWARE. ZDROJ [VANÍČEK 2004A].	19
OBRÁZEK 3: KVALITA A ŽIVOTNÍ CYKLUS SOFTWARE. ZDROJ [ISO/IEC 2009A].	22
OBRÁZEK 4: OBECNÝ REFERENČNÍ MODEL SQUARE. ZDROJ [ISO/IEC 2004].	31
OBRÁZEK 5: ORGANIZACE MEZINÁRODNÍCH NOREM SQUARE. ZDROJ [ISO/IEC 2009A].	32
OBRÁZEK 6: PŘEHLED STAVU PROJEKTU SQUARE. ZDROJ [VANÍČEK 2008].	33
OBRÁZEK 7: STRUKTURA QUALITY MODEL DIVISION. ZDROJ [ISO/IEC 2011A].	39
OBRÁZEK 8: VZTAH ISO/IEC 25021 JAKO SPOJITOST MEZI SÉRIÍ 9126 A ŘADOU NOREM SQUARE ZDROJ [ISO/IEC 2011A].	39
OBRÁZEK 9: CELKOVÝ RÁMEC HODNOCENÍ JAKOSTI PRODUKTŮ. ZDROJ [ISO/IEC 2011B].	42
OBRÁZEK 10: STRUKTURA MODELU KVALITY. ZDROJ [ISO/IEC 2009A].	45
OBRÁZEK 11: MODEL KVALITY UŽITÍ SYSTÉMU. ZDROJ [ISO/IEC 2009A].	45
OBRÁZEK 12: MODEL KVALITY SOFTWAREOVÉHO PRODUKTU. ZDROJ [ISO/IEC 2009A].	46
OBRÁZEK 13: PŘÍKLAD MODELU SYSTÉMU A MODELŮ KVALITY. ZDROJ [ISO/IEC 2006A].	48
OBRÁZEK 14: REFERENČNÍ MODEL MĚŘENÍ KVALITY SOFTWAREOVÉHO PRODUKTU. ZDROJ [ISO/IEC 2009A].	52
OBRÁZEK 15: VLASTNOSTI SOFTWARE. ZDROJ: [ISO/IEC 2006A].	53
OBRÁZEK 16: VYMEZENÍ A ANALÝZA POŽADAVKŮ NA KVALITU SOFTWARE. ZDROJ: [ISO/IEC 2006A].	55
OBRÁZEK 17: KATEGORIZACE POŽADAVKŮ NA SYSTÉM. [ISO/IEC 2006A].	56
OBRÁZEK 18: DEFINICE A ANALÝZA POŽADAVKŮ ZÚČASTNĚNÝCH STRAN. [ISO/IEC 2006A].	58
OBRÁZEK 19: HIERARCHIE - SYSTÉM A POŽADAVKY NA SOFTWARE. ZDROJ [ISO/IEC 2006A].	59
OBRÁZEK 20: REFERENČNÍ MODEL HODNOCENÍ KVALITY SOFTWAREOVÉHO. ZDROJ [VANÍČEK 2008C].	60
OBRÁZEK 21: OBECNÝ POSTUP AUDITU. ZDROJ [SVATÁ 2011].	72
OBRÁZEK 22: VZTAH MEZI KVANTIFIKOVANOU VLASTNOSTÍ, METODOU MĚŘENÍ A QME. ZDROJ [ISO/IEC 2011A].	78
OBRÁZEK 23: VZTAH MEZI VLASTNOSTÍ KE KVANTIFIKACI, METODOU MĚŘENÍ, QME A METODOU MĚŘENÍ. ZDROJ [ISO/IEC 2011A].	79
OBRÁZEK 24: STRUKTURA METODY PRO KVANTIFIKOVANÝ PŘÍSTUP K JAKOSTI IS. ZDROJ: AUTOR.	98
OBRÁZEK 25: PŘÍŘAZENÍ PRIORIT QME. ZDROJ: AUTOR.	119

Seznam tabulek

TABULKA 1: ROZDÍLY MEZI MÍRAMI INTERNÍ JAKOSTI, EXTERNÍ JAKOSTI, A JAKOSTI UŽITÍ. ZDROJ: [ISO/IEC 2009A].	54
TABULKA 2: FORMÁT METODY MĚŘENÍ PRO QME. ZDROJ: [ISO/IEC 2011A].	80
TABULKA 3: ZÁKLADNÍ SET QME. ZDROJ: [ISO/IEC 2011A].	82
TABULKA 4: MAPOVÁNÍ MEZI POČÁTEČNÍ QME A JEJICH POUŽITÍM PRO FUNKČNÍ PŘIMĚŘENOST, KOMPATIBILITU, POUŽITELNOST A ÚČINNOST DLE ISO/IEC 25010. ZDROJ: [ISO/IEC 2011A].	84

TABULKA 5: MAPOVÁNÍ MEZI POČÁTEČNÍ QME A JEJICH POUŽITÍM PRO BEZPORUCHOVOST, BEZPEČNOST, UDRŽOVATELNOST A PŘENOSITELNOST DLE ISO/IEC 25010. ZDROJ: [ISO/IEC 2011A].	85
TABULKA 6: ZÁKLADNÍ FAKTA PRO OVĚŘENÍ NAVRHOVANÉ METODY. ZDROJ: AUTOR.	99
TABULKA 7: MAPOVÁNÍ MEZI QME A JEJICH POUŽITÍM PRO FUNKČNÍ PŘÍMĚŘENOST, KOMPATIBILITU, POUŽITELNOST A ÚČINNOST DLE APLIKAČNÍHO MANAŽERA. ZDROJ: AUTOR.	101
TABULKA 8: MAPOVÁNÍ MEZI QME A JEJICH POUŽITÍM PRO BEZPORUCHOVOST, BEZPEČNOST, UDRŽOVATELNOST A PŘENOSITELNOST DLE APLIKAČNÍHO MANAŽERA. ZDROJ: AUTOR.	103
TABULKA 9: NAVRŽENÉ QME DLE POŽADAVKŮ APLIKAČNÍHO MANAŽERA. ZDROJ: AUTOR.	104
TABULKA 10: MAPOVÁNÍ MEZI QME A JEJICH POUŽITÍM PRO FUNKČNÍ PŘÍMĚŘENOST, KOMPATIBILITU, POUŽITELNOST A ÚČINNOST DLE ICT DEVELOPERA. ZDROJ: AUTOR.	106
TABULKA 11: MAPOVÁNÍ MEZI QME A JEJICH POUŽITÍM PRO BEZPORUCHOVOST, BEZPEČNOST, UDRŽOVATELNOST A PŘENOSITELNOST DLE ICT DEVELOPERA. ZDROJ: AUTOR.	108
TABULKA 12: NAVRŽENÉ QME DLE POŽADAVKŮ ICT DEVELOPERA. ZDROJ: AUTOR.	110
TABULKA 13: MAPOVÁNÍ MEZI QME A JEJICH POUŽITÍM PRO FUNKČNÍ PŘÍMĚŘENOST, KOMPATIBILITU, POUŽITELNOST A ÚČINNOST DLE BUSINESS ARCHITEKTA. ZDROJ: AUTOR.	111
TABULKA 14: MAPOVÁNÍ MEZI QME A JEJICH POUŽITÍM PRO BEZPORUCHOVOST, BEZPEČNOST, UDRŽOVATELNOST A PŘENOSITELNOST DLE BUSINESS ARCHITEKTA. ZDROJ: AUTOR.	112
TABULKA 15: PŘÍRAZENÍ PRIORIT QME PODLE ODHADU APLIKAČNÍHO MANAŽERA. ZDROJ: AUTOR.	113
TABULKA 16: PŘÍRAZENÍ PRIORIT QME PODLE ODHADU ICT DEVELOPERA. ZDROJ: AUTOR.	114
TABULKA 17: PŘÍRAZENÍ PRIORIT QME PODLE ODHADU BUSINESS ARCHITEKTA. ZDROJ: AUTOR.	115
TABULKA 18: PŘÍRAZENÍ PRIORIT NOVĚ DOPLNĚNÝM QME. ZDROJ: AUTOR.	115
TABULKA 19: VÝSLEDNÝ MODEL QME. ZDROJ: AUTOR.	117

Seznam zkratk

COBIT	Control Objectives for Information and related Technology
ČSNI	Český normalizační institut
HW	hardware
IS/IT	informační systém/informační technologie
ICT	informační a komunikační technologie
INTOSAI	International Organisation of Supreme Audit Institutions
ISACA	Information System Audit and Control Association
ISO/IEC	International Organization for Standardization/International Electrotechnical Commission
JCT	Joint Technical Committee
QM	Quality Measure
QME	Quality Measure Element
SQUARE	Software Quality Requirements and Evaluation
SLA	Service Level Agreement

SW	software
TNK	Technická normalizační komise
WG	Work Group

Literatura a zdroje

- [Basl 2002] BASL, Josef. *Podnikové informační systémy. Podnik v informační společnosti*. Praha: Grada Publishing, s.r.o., 2002. 144 s. ISBN 80-247-0214-2.
- [BSC 2010] *Balanced Scorecard Institute. What is the Balanced Scorecard?* [online]. c1998-2010, [cit. 2010-03-14]. Dostupné z: <<http://www.balancedscorecard.org/BSCResources/AbouttheBalancedScorecard/tabid/55/Default.aspx>>
- [COBIT 2007] *IT Governance. Co je COBIT?*. [online]. c2007, [cit. 2010-03-20]. Dostupné z: <<http://www.itil.cz/index.php?id=929>>
- [Dohnal, Příklenk 2011] DOHNAL J.- Příklenk O. *CIO a podpora byznysu případovými studii CIO v ČR a SR*. Praha: Grada Publishing, a.s, 2011. 176 s. ISBN 978-80-247-4050-8.
- [EVAL 2006] *Research Methods Knowledge Base. Introduction to Evaluation*. [online]. c2006, [cit. 2012-01-07]. Dostupné z: <<http://www.socialresearchmethods.net/kb/intreval.php>>
- [Gála et al. 2009] GÁLA L.- POUR J.- ŠEDIVÁ Z. *Podniková informatika. 2., přepracované a aktualizované vydání*. Praha: Grada Publishing, a.s, 2009. 496 s. ISBN 978-80-247-2615-1.
- [INFO 2008] UNICORN Systems, a.s. *Informační systém v oblasti bankovníctví*. [online]. c2008, [cit. 2012-01-07]. Dostupné z: <<http://www.unicornsyste.ms.eu/cz/press/clanky/cla080512.html>>
- [ISO/IEC 2004] *ISO/IEC FCD 25000: Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE*. Department of Industrial and Management Systems Eng. Waseda University, Tokyo, Japan, 2004.
- [ISO/IEC 2006a] *ISO/IEC CD 25030: Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) – Quality requirements*. Department of Industrial and Management Systems Eng. Waseda University, Tokyo, Japan, 2006.
- [ISO/IEC 2006b] *ISO/IEC 25051: Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) – Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing*. 2006.
- [ISO/IEC 2009a] *ISO/IEC FCD 25010: Systems and software engineering – System and software product Quality Requirements and Evaluation (SQuaRE) – System and software quality models*. Department of Industrial and Management Systems Eng. Waseda University, Tokyo, Japan, 2009.
- [ISO/IEC 2009b] *International Organization for Standardization. ISO/IEC JTC 1 Software and Systems Engineering* [online]. c2009, [cit. 2010-03-20]. Dostupné z: <<http://www.jtc1-sc7.org/>>
- [ISO/IEC 2010] *International Organization for Standardization. ISO/IEC JTC 1 Information Technology Standards* [online]. c2010, [cit. 2010-03-20]. Dostupné z: <http://www.iso.org/iso/jtc1_home.html>

- [ISO/IEC 2011a] *ISO/IEC DIS 25021: Systems and software engineering – System and software product Quality Requirements and Evaluation (SQuaRE) – Quality measure elements*. Department of Industrial and Management Systems Eng. Waseda University, Tokyo, Japan, 2011.
- [ISO/IEC 2011b] *ISO/IEC DIS 25041: Systems and software engineering – System and software product Quality Requirements and Evaluation (SQuaRE) – Evaluation guide for developers, acquirers and independent evaluators*. Department of Industrial and Management Systems Eng. Waseda University, Tokyo, Japan, 2011.
- [ISO/IEC 2011c] *ISO/IEC WD 25023: Systems and software engineering – System and software product Quality Requirements and Evaluation (SQuaRE) – Measurement of system and software product quality*. Department of Industrial and Management Systems Eng. Waseda University, Tokyo, Japan, 2011.
- [Kardoš 2010] KARDOŠ, Daniel. *Statický a dynamický přístup k jakosti informačních systémů*. Praha, 2010. 120 s. Disertační práce na Provozně ekonomické fakultě České zemědělské univerzity v Praze na katedře informačního inženýrství. Vedoucí disertační práce Prof. RNDr. Jiří Vaníček, CSc.
- [Král et al. 2006] KRÁL, Bohumil a kol. *Manažerské účetnictví. 2. rozšířené vydání*. Praha: Management Press, s.r.o., 2006. 622 s. ISBN 80-7261-141-0.
- [Lacko 2010] LACKO, Branislav. *Expertní odhady v projektech tvorby software*. In TVORBA SOFTWARE 2010. Tvorba softwaru a programování Ostrava. [online], [cit. 2011-12-18]. Dostupné z: <<http://formular-ekf.vsb.cz/formulare/F01/tsw/?page=prispevek&rok=2010&prispevekid=978>>
- [Molnár 2000] MOLNÁR, Zdeněk. *EFEKTIVNOST IS/IT* [online]. c2000, [cit. 2010-03-15]. Dostupné z: <http://gis.vsb.cz/GIS_Ostrava/GIS_Ova_2000/Sbornik/Molnar/Referat.htm>
- [Molnár et al. 1999] MOLNÁR, Z., PAVLÍK, D., VLASÁK, J. *Efektivnost IS/IT v podnicích České republiky*. [online]. c1999, [cit. 2010-03-20]. Dostupné z: <<http://si.vse.cz/archive/proceedings/1999/efektivnost-is-it-v-podnicich-cr.pdf>>
- [Novotný et al. 2011] NOVOTNÝ, Ota a kol. *Řízení výkonnosti podnikové informatiky*. Praha: Professional Publishing, 2011. 276 s. ISBN 978-80-7431-040-9.
- [Svatá 2011] SVATÁ, Vlasta. *Audit informačního systému*. Praha: Profesional Publishing, 2011. 219 s. ISBN 978-80-7431-034-8.
- [Synek et al. 2006] SYNEK, Miloslav a kol. *Podniková ekonomika. 4. přepracované a doplněné vydání*. Praha: C. H. Beck, 2006. 475 s. ISBN 80-7179-892-4.
- [Synek et al. 2007] SYNEK, Miloslav a kol. *Manažerská ekonomika. 4., aktualizované a rozšířené vydání*. Praha: Grada Publishing, a.s., 2007. 464 s. ISBN 978-80-247-1992-4.
- [Tvrdíková 2000] TVRDÍKOVÁ, Milena. *Zavádění a inovace informačních systémů ve firmách*. Praha: Grada Publishing, s.r.o., 2000. 116 s. ISBN 80-7169-703-6.
- [Učeň et al. 2001] UČEŇ, Pavel a kol. *Metriky v informatice. Jak objektivně zjistit přínosy informačního systému*. Praha: Grada Publishing, s.r.o., 2001. 140 s. ISBN 80-247-0080-8.
- [Vaníček 2004a] VANÍČEK, Jiří. *Měření a hodnocení jakosti informačních systémů*. Praha: PEF-ČZU v Praze, 2004. 328 s. ISBN 80-213-1206-8.
- [Vaníček 2004b] VANÍČEK, Jiří. *Kvalita software ve světle mezinárodních norem*. In TVORBA SOFTWARE 2004. Tvorba softwaru a programování Ostrava. [online], [cit. 2011-12-18]. Dostupné z: <<http://projects.cba.muni.cz/tsw/?page=prispevek&rok=2004&prispevekid=818>>
- [Vaníček 2006] VANÍČEK, Jiří. *Software quality requirements*. [online], [cit. 2011-12-18]. Dostupné z: <http://www.cazv.cz/attachments/ZE_52_177-185.pdf>
- [Vaníček 2008a] VANÍČEK, Jiří. *Mezinárodní normalizace kvality software*. In OBJEKTY 2008. *Objekty v škole vs. Objekty v praxi*. Žilina: Katedra softvérových technologií. Katedra riadenia a informatiky. Žilinská univerzita v Žilíně. 2008. ISBN 978-80-8070-927-3.

- [Vaniček 2008b] VANÍČEK, Jiří. *Mezinárodní normy řady ISO/IEC 250xx pro jakost softwarového produktu*. Linux+, 41, 2008 (2) s. 68 – 74, ISSN 1733-4209
- [Vaniček 2008c] VANÍČEK, Jiří. *Kvalita informačních systémů z pohledu mezinárodní normalizace (aktuální informace)*. [online], [cit. 2011-12-18]. Dostupné z: <http://czu.vasekk.cz/Ing/04_desaty%20semestr/Jakost_IS/08_SQuaRE.pdf>
- [Vaniček 2010] VANÍČEK, Jiří. *Information systems quality rating*. Praha: PEF-ČZU v Praze, 2010. 70s. ISBN 978-80-213-2026-8
- [Vrana, Richta 2005] VRANA I.- RICHTA K. *Zásady a postupy zavádění podnikových informačních systémů. Praktická příručka pro manažery*. Praha: Grada Publishing, s.r.o., 2005. 188 s. ISBN 80-247-1103-6.
- [Vrana et al. 2012] VRANA, I., et al. *Afuzzy group agreement-based approach for multiexpert decision making in environmental issues*. Environmental Modelling & Software (2012), doi:10.1016/j.unvsoft.2011.12.007.
- [Wagner 2009] WAGNER, Jaroslav. *Měření výkonnosti. Jak měřit, vyhodnocovat a využívat informace o podnikové výkonnosti*. Praha: Grada Publishing, a.s., 2009. 256 s. ISBN 978-80-247-2924-4.

Příloha A - Návrh prvků měření kvality

a) Název QME	Soulad dokumentace a software
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry je obsah dokumentace k informačnímu systému konzistentní k implementovanému produktu.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet bloků informačního systému, které jsou úplně a přesně popsány B = počet bloků informačního systému
e) Metoda měření	Z dokumentace k softwarovému produktu definovat počet popsaných (požadovaných) bloků informačního systému a následně zkoumáním porovnat s implementovaným software.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Blok informačního systému Systémová dokumentace
g) Vymezení jednotlivých dílčích vlastností	Blok informačního systému: modul, funkční část systému. Systémová dokumentace: odsouhlasený a potvrzený soubor dokumentů popisující implementovaný produkt.
h) Vstup pro QME	Vstupem je vymezení a počet bloků informačního systému, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet bloků (modulů) systému
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření implementace a celkového fungování informačního systému dle zákaznických požadavků, a současně k prověření kvality systémové dokumentace.
m) Software životního cyklu procesu (ú)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná dokumentace k software, včetně funkční specifikace a smluvních ujednání.

a) Název QME	Předpoklad shody
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je porovnat očekávané výstupy systému s aktuálními výstupy z informačního systému.
d) Relevantní míra(y) kvality	$X = A/t$ A = počet případů, které přesáhly rozdíl mezi aktuálním a očekávaným výsledkem t = čas běhu systému
e) Metoda měření	Sledovat za určené období běhu informačního systému a zaznamenávat ty případy, kdy došlo k rozdílu mezi očekávaným a reálným výstupem.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Výsledek

g) Vymezení jednotlivých dílčích vlastností	Výsledek: výstup systému po provedení operace nebo aktivity
h) Vstup pro QME	Vstupem je vymezení přípustné odchylky mezi očekávaným a skutečným výstupem, a současně vymezení sledovaného období běhu informačního systému.
i) Jednotka měření pro QME	$0 < X$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 0, tím lépe
j) Numerická pravidla	Počet neshodných výstupů systému Vymezení sledovaného období
k) Typ měřicí stupnice	Poměrová
l) Kontext QME	Toto QME slouží k ověření přesnosti výstupů dodávaných implementovaným software.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici přehled výstupů systému, které byly identifikovány jako neshodné s očekávaným výstupem systému za sledované období.

a) Název QME	Obnovitelnost systému
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit schopnost obnovy systému po abnormální události.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet případů úspěšné obnovy systému B = počet případů sledovaných abnormálních událostí
e) Metoda měření	Z dokumentace k softwarovému produktu definovat počet popsanych (požadovaných) bloků informačního systému a následně zkoumáním porovnat s implementovaným software
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Abnormální událost
g) Vymezení jednotlivých dílčích vlastností	Abnormální událost: nestandardní stav chování systému.
h) Vstup pro QME	Vstupem je vymezení abnormální události a zaznamenaný počet výskytu těchto událostí.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet výskytu abnormálních událostí za sledované období.
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření schopnosti obnovy informačního systému při výskytu abnormálních událostí.
m) Software životního cyklu procesu (ů)	Provoz

n) Omezení měření	Pro měření musí být k dispozici přehled počtu případů úspěšné obnovy systému a současně přehled počtu případů abnormálních událostí za sledované období.
--------------------------	--

a) Název QME	Nedostupnost systému
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit průměrnou dobu nedostupnosti informačního systému, pokud dojde k jeho selhání, do doby jeho postupného spuštění.
d) Relevantní míra(y) kvality	$X = t/A$ t = celkový čas nedostupnosti systému A = počet zaznamenaných zhroucení systému
e) Metoda měření	Změřit a zaznamenat počet zhroucení systému a současně dobu, po kterou nebyl systém dostupný.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Zhroucení systému
g) Vymezení jednotlivých dílčích vlastností	Zhroucení systému: stav, kdy systém není k dispozici a neplní svoje funkce. Čas nedostupnosti systému: doba celkové požadované délky provozu.
h) Vstup pro QME	Vstupem pro QME je čas, po který systém není k dispozici a neplní tak svoje funkce.
i) Jednotka měření pro QME	$0 < X$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 0, tím lépe
j) Numerická pravidla	Počet zaznamenaných zhroucení systému
k) Typ měřicí stupnice	Poměrová
l) Kontext QME	Toto QME slouží k prověření implementace a celkového fungování informačního systému dle zákaznických požadavků, a současně k prověření kvality systémové dokumentace.
m) Software životního cyklu procesu (ú)	Provoz
n) Omezení měření	Pro měření musí být k dispozici zaznamenaný počet a doba, kdy byl systém nedostupný.

a) Název QME	Modifikace kódu
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, zda a do jaké míry funkcionality systému mohou být upraveny pomocí modifikace kódu během údržby systému.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet funkcí informačního systému, které mohou být udržovány úpravou kódu B = počet funkcí informačního systému, které mají být upraveny
e) Metoda měření	Prověřit u dodavatele/vývoje možnost modifikace kódu. Dle výsledku pak prověřit možnost modifikace v rámci údržby systému a na základě těchto informací provést měření.
f) Seznam sub vlastností souvisejících s vlastností	Funkce systému

ke kvantifikaci	
g) Vymezení jednotlivých dílčích vlastností	Funkce systému, která má být upravena: customizace.
h) Vstup pro QME	Vstupem je vymezení a počet bloků informačního systému, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet zaznamenaných funkcí, které mají být upraveny.
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k zjištění rozsahu využití vlastních IT zdrojů.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná funkční specifikace a možnost zásahu do zdrojového kódu. Tato možnost je smluvně ošetřena mezi dodavatelem a provozovatelem systému.

a) Název QME	Počet funkčních požadavků
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry jsou realizovány specifikované funkční požadavky na informační systém.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet realizovaných funkčních požadavků B = počet požadovaných funkčních požadavků
e) Metoda měření	Z dokumentace k softwarovému produktu definovat počet popsanych a dohodnutých (požadovaných) funkčních požadavků na informační systém a následně zkoumáním porovnat s implementovaným software.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Funkční požadavek na IS
g) Vymezení jednotlivých dílčích vlastností	Funkční požadavek na IS: dle specifikace
h) Vstup pro QME	Vstupem je počet funkčních požadavků na informační systém, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet schválených funkčních požadavků na IS
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření implementace a celkového fungování informačního systému dle zadaných funkčních požadavků, a současně k prověření kvality systémové dokumentace.
m) Software životního cyklu procesu (ů)	Provoz

n) Omezení měření	Pro měření musí být k dispozici úplná dokumentace k software, včetně funkční specifikace.
--------------------------	---

a) Název QME	Počet systémových požadavků
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry jsou realizovány specifikované systémové požadavky na informační systém.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet realizovaných systémových požadavků B = počet požadovaných systémových požadavků
e) Metoda měření	Z dokumentace k softwarovému produktu definovat počet popsanych a dohodnutých (požadovaných) systémových požadavků na informační systém a následně zkoumáním porovnat s implementovaným software.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Systémový požadavek na IS
g) Vymezení jednotlivých dílčích vlastností	Systémový požadavek na IS: dle specifikace.
h) Vstup pro QME	Vstupem je počet systémových požadavků na informační systém, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet schválených systémových požadavků na IS.
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření implementace a celkového fungování informačního systému dle zadaných systémových požadavků, a současně k prověření kvality systémové dokumentace.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná dokumentace k software, včetně funkční specifikace.

a) Název QME	Počet hesel uživatelské dokumentace
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry je detailně a srozumitelně vytvořená uživatelská dokumentace pro pochopení uživatelem.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet aktivit, které lze správně provést po přečtení dokumentace B = počet aktivit uživatelské dokumentace
e) Metoda měření	Z uživatelské dokumentace k softwarovému produktu definovat počet popsanych aktivit informačního systému a následně zkoumáním porovnat se zkušenostmi na vybraném vzorku uživatelů.
f) Seznam sub vlastností	Aktivita

souvisejících s vlastností ke kvantifikaci	
g) Vymezení jednotlivých dílčích vlastností	Aktivita: operace/sled operací vedoucích k výsledku (výstupu) systému.
h) Vstup pro QME	Vstupem je vymezení a počet aktivit, které uživatel může provádět při správném používání produktu po přečtení dokumentace.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet aktivit, které uživatel provede v systému.
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření kvality uživatelské dokumentace produktu.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná uživatelská dokumentace k software.

a) Název QME	Počet úkolů (úkonů) popsaných v uživ. nápovědě
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry je srozumitelná uživatelská nápověda v informačním systému při jeho správném používání.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet úkonů, které lze provést po přečtení nápovědy B = počet všech úkonů v uživatelské nápovědě
e) Metoda měření	Z dokumentace k softwarovému produktu definovat počet posaných (požadovaných) úkonů v uživatelské nápovědě informačního systému a následně zkoumáním porovnat se zkušenostmi na vybraném vzorku uživatelů.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Úkon v uživatelské nápovědě
g) Vymezení jednotlivých dílčích vlastností	Úkon v uživatelské nápovědě: přesně definovaná a popsaná aktivita/krok procesu, kterou uživatel může v IS provést.
h) Vstup pro QME	Vstupem je vymezení a počet úkonů, které jsou obsahem uživatelské nápovědy informačního systému.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet úkonů v uživatelské nápovědě
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření kvality uživatelské nápovědy produktu.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná dokumentace k software, včetně funkční specifikace.

a) Název QME	Počet hesel administrátorské dokumentace
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry je detailně a srozumitelně vytvořená administrátorská dokumentace.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet aktivit, které lze správně provést po přečtení dokumentace B = počet aktivit administrátorské dokumentace
e) Metoda měření	Z administrátorské dokumentace k softwarovému produktu definovat počet popsaných aktivit informačního systému a následně zkoumáním porovnat se zkušenostmi na vybraném vzorku uživatelů.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Aktivita
g) Vymezení jednotlivých dílčích vlastností	Aktivita: operace/sled operací vedoucích k výsledku při administraci systému
h) Vstup pro QME	Vstupem je vymezení a počet aktivit, které administrátor může provádět po přečtení dokumentace.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet aktivit, které administrátor provede v systému.
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření kvality administrátorské dokumentace produktu.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná administrátorská dokumentace včetně funkční specifikace k software.

a) Název QME	Počet vstupních/import formátů dat
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je porovnat implementované formáty vstupních dat s požadavky na import dat do systému.
d) Relevantní míra(y) kvality	$X = (A/B)*100$ A = realizovaný počet vstupních formátů dat B = požadovaný počet vstupních formátů dat
e) Metoda měření	Z dokumentace k softwarovému produktu definovat počet popsaných (požadovaných) vstupních formátů dat informačního systému a následně zkoumáním porovnat s implementovaným software.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Vstupní formát dat
g) Vymezení jednotlivých dílčích vlastností	Vstupní formát dat

h) Vstup pro QME	Vstupem je vymezení a počet vstupních formátů dat informačního systému, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu.
i) Jednotka měření pro QME	$0 \leq X \leq 100$ Procenta, čím blíže je poměrové číslo k 100, tím lépe
j) Numerická pravidla	Počet vstupních formátů dat
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření implementace a celkového fungování informačního systému dle zákaznických požadavků.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná dokumentace k software, včetně funkční specifikace.

a) Název QME	Počet výstupních/export formátů dat
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je porovnat implementované formáty výstupních dat s požadavky na export dat ze systému.
d) Relevantní míra(y) kvality	$X = (A/B) * 100$ A = realizovaný počet výstupních h formátů dat B = požadovaný počet výstupních formátů dat
e) Metoda měření	Z dokumentace k softwarovému produktu definovat počet popsaných (požadovaných) výstupních formátů dat informačního systému a následně zkoumáním porovnat s implementovaným software.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Výstupní formát dat
g) Vymezení jednotlivých dílčích vlastností	Výstupní formát dat
h) Vstup pro QME	Vstupem je vymezení a počet výstupních formátů dat informačního systému, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu.
i) Jednotka měření pro QME	$0 \leq X \leq 100$ Procenta, čím blíže je poměrové číslo k 100, tím lépe
j) Numerická pravidla	Počet výstupních formátů dat
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření implementace a celkového fungování informačního systému dle zákaznických požadavků.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná dokumentace k software, včetně funkční specifikace.

a) Název QME	Počet doplňků, pluginů
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry jsou realizovány specifikované doplňky na informační systém.
d) Relevantní míra(y) kvality	$X = A/B$ A = počet realizovaných doplňků B = počet požadovaných doplňků systému
e) Metoda měření	Z dokumentace k softwarovému produktu definovat počet popsanych (požadovaných) doplňků informačního systému a následně zkoumáním porovnat s implementovaným software.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Doplňek (plugin)
g) Vymezení jednotlivých dílčích vlastností	Doplňek (plugin): modul, nastavbová část software, která zajišťuje specifické funkcionality.
h) Vstup pro QME	Vstupem je vymezení a počet pluginů informačního systému, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu.
i) Jednotka měření pro QME	$0 \leq X \leq 1$ Bezrozměrná jednotka, čím blíže je poměrové číslo k 1, tím lépe
j) Numerická pravidla	Počet doplňků (pluginů) systému.
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření implementace doplňků a celkového fungování informačního systému dle zákaznických požadavků.
m) Software životního cyklu procesu (ů)	Provoz
n) Omezení měření	Pro měření musí být k dispozici úplná dokumentace k software, včetně funkční specifikace a smluvních ujednání.

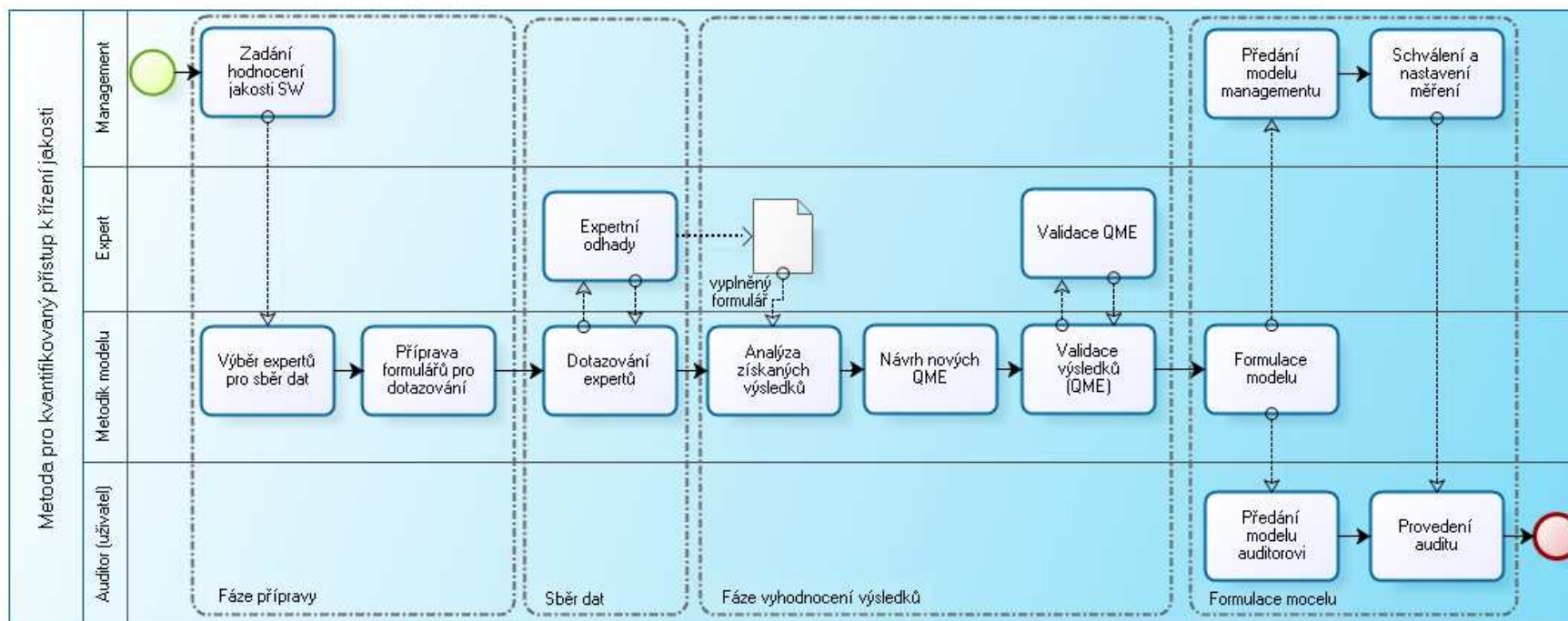
a) Název QME	Počet použitých analytických patternů
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry byly použity standardní analytické best-practice postupy.
d) Relevantní míra(y) kvality	$X = (A/B)*100$ A = počet použitých analytických patternů B = počet použitých postupů
e) Metoda měření	Z analytické dokumentace k softwarovému produktu zjistit a prověřit použité postupy pro analytickou fázi vývoje informačního systému.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Analyticý pattern
g) Vymezení jednotlivých dílčích vlastností	Analytické patterny: vhodné postupy, které pomáhají analýzu posunout na vyšší kvalitativní úroveň

h) Vstup pro QME	Vstupem je vymezení a počet použitých analytických patternů, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu
i) Jednotka měření pro QME	$0 \leq X \leq 100$ Procenta, čím blíže je poměrové číslo k 100, tím lépe
j) Numerická pravidla	Počet analytických patternů
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření analytických postupů uplatněných při vývoji softwarového produktu.
m) Software životního cyklu procesu (ů)	Vývoj, provoz
n) Omezení měření	Pro měření musí být k dispozici úplná analytická dokumentace k software, včetně funkční specifikace

a) Název QME	Počet použitých vývojářských patternů
b) Cílová entita	Aplikační software
c) Cíle a vlastnosti ke kvantifikaci	Cílem je zjistit, do jaké míry byly použity standardní vývojářské best-practice postupy.
d) Relevantní míra(y) kvality	$X = (A/B) * 100$ A = počet použitých analytických patternů B = počet použitých postupů
e) Metoda měření	Z analytické dokumentace k softwarovému produktu zjistit a prověřit použité postupy pro fázi vývoje informačního systému.
f) Seznam sub vlastností souvisejících s vlastností ke kvantifikaci	Vývojářský pattern
g) Vymezení jednotlivých dílčích vlastností	Vývojářské patterny: návrhové vzory pro řešení problémů pro efektivní design software
h) Vstup pro QME	Vstupem je vymezení a počet použitých vývojářských patternů, který je porovnáván dle dokumentace (požadavků) a na implementovaném produktu
i) Jednotka měření pro QME	$0 \leq X \leq 100$ Procenta, čím blíže je poměrové číslo k 100, tím lépe
j) Numerická pravidla	Počet vývojářských patternů
k) Typ měřicí stupnice	Absolutní
l) Kontext QME	Toto QME slouží k prověření vývojářských postupů uplatněných při vývoji softwarového produktu.
m) Software životního cyklu procesu (ů)	Vývoj, provoz
n) Omezení měření	Pro měření musí být k dispozici úplná analytická dokumentace k software, včetně funkční specifikace.

Příloha B

Návrh metody v notaci BPMN (*Business Process Modeling Notation*)⁴



⁴ Pro tvorbu obrázku autor použil volně dostupný software Bizagi. Zdroj software: www.bizagi.com.