

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

ANALÝZA RIZIK V MANAGEMENTU PROJEKTŮ

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. JÁN PAULENDA

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

ANALÝZA RIZIK V MANAGEMENTU PROJEKTŮ

RISKS ANALYSIS IN PROJECT MANAGEMENT

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. JÁN PAULENDA

VEDOUCÍ PRÁCE
SUPERVISOR

doc. RNDr. JITKA KRESLÍKOVÁ, CSc.

BRNO 2011

Abstrakt

Diplomová práce se zabývá problematikou managementu rizik s důrazem na etapu analýzy rizik. V rámci managementu rizik jsou popsány všechny důležité etapy, pojmy a techniky. V kontextě analýzy rizik je představeno několik používaných metod, přičemž je kladen důraz hlavně na Markovovu analýzu a analýzu pomocí Petriho sítí. Po důkladné analýze problematiky byl navržen a implementován systém pro podporu managementu rizik, přičemž byla zvolena právě Markovova analýza.

Abstract

The main subject of this Master's thesis is risk management with emphasis on risk analysis stage. All important stages, definitions and techniques are introduced in the context of risk management. There are several risk analysis methods discussed in this thesis, especially the Markov analysis and the Petri net analysis. After the thorough analysis of the subject, the system to support the risk management was designed and implemented, while precisely Markov analysis was chosen.

Klíčová slova

riziko, management rizik, analýza rizik, Markovova analýza, Petriho síť

Keywords

risk, risk management, risk analysis, Markov analysis, Petri net

Citace

Ján Paulenda: Analýza rizik v managementu projektů, diplomová práce, Brno, FIT VUT v Brně, 2011

Analýza rizik v managementu projektů

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením paní doc. RNDr. Jitky Kreslíkové, CSc.

.....

Ján Paulenda

24. května 2011

Poděkování

Ďakujem mojej vedúcej – doc. RNDr. Jitke Kreslíkovej, CSc. za podporu, cenné poznatky a pomoc pri získavaní zdrojov, potrebných na vypracovanie tejto diplomovej práce.

© Ján Paulenda, 2011.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod	3
2 Manažment projektov	4
2.1 Projekt	4
2.2 Projektový manažment	5
2.3 Manažment projektov všeobecne	6
2.4 Manažment IT projektov	6
2.5 Zainteresované subjekty	7
2.6 Analýza rizík v manažmente projektov	7
3 Manažment rizík	9
3.1 Riziko a príležitosť	9
3.2 Štruktúra manažmentu rizík	9
3.3 Rizikový posudok a jeho miesto v manažmente rizík	10
3.3.1 Komunikácia a konzultácia	11
3.3.2 Ustanovenie kontextu	12
3.3.3 Rizikový posudok	12
3.3.4 Spracovanie rizík	13
3.3.5 Kontrola a zhodnotenie	16
3.4 Proces posúdenia rizík	16
3.4.1 Identifikácia rizík	17
3.4.2 Analýza rizík	19
3.4.3 Vyhodnotenie rizík	20
3.5 Techniky a metódy analýzy rizík	21
3.5.1 Modelovanie a simulácia	22
3.5.2 Riadená diskusia o riešení nových problémov (brainstorming)	23
3.5.3 Štrukturovaná „What if“– technika	23
3.5.4 Analýza stromom udalostí	23
3.5.5 Analýza rozhodovacím stromom	24
3.5.6 Simulácia Monte Carlo	25
3.5.7 Markovova analýza	25
3.5.8 Analýza Petriho sieťami	29
3.6 Markovova analýza verzus analýza Petriho sieťami	32
3.6.1 Aplikácia modelovacích techník v manažmente rizík IT projektov	32
3.6.2 Zvolenie modelovacej techniky na implementáciu aplikácie	33

4	Analýza a návrh aplikácie	34
4.1	Neformálna špecifikácia	34
4.2	Spôsob uloženia dát	35
4.2.1	Štruktúra dokumentu v jazyku XML	35
4.2.2	Návrh XML jazyka pre vyvíjanú aplikáciu	36
4.3	Prípady použitia	37
4.4	Návrh grafického rozhrania	37
5	Implementácia	38
5.1	Implementačné nástroje	38
5.1.1	Jazyk JAVA	38
5.1.2	NetBeans IDE 6.8	39
5.1.3	Knižnica JfreeChart	39
5.1.4	Knižnica dom4j	39
5.2	Hlavné funkcie aplikácie	39
5.2.1	Hlavné okno a kalkulácia Markovovho reťazca	40
5.2.2	Vykreslenie grafu	43
5.2.3	Generovanie Markovovho diagramu	44
5.3	Ilustrácia funkčnosti aplikácie na vybraných Markovových modeloch	46
6	Záver	51
6.1	Ďalší rozvoj aplikácie	51
A	Obsah DVD	55

Kapitola 1

Úvod

V každodenných procesoch plánovania a vykonávania nejakých činností je takmer vždy prítomný aspekt rizika. Mnohé z nich majú negatívny dopad na širokú škálu vecí, či už sa jedná o zdravie ľudí, financie, prípadne čas alebo iné hmotné a nehmotné atribúty. Tento fakt si začalo uvedomovať čoraz viac organizácií, v ktorých sa dnes už takmer spravidla vyčleňuje priestor pre tzv. Manažment rizík.

Manažment rizík je súčasťou manažmentu projektov, ktorému sa venuje druhá kapitola. V nej je kladený dôraz na najdôležitejšie pojmy tejto oblasti, aby bolo neskôr možné plynulo prejsť k samotnému manažmentu rizík.

Tomu sa už venuje celá tretia kapitola, ktorá obsahuje množstvo normovaných definícií rôznych pojmov a vysvetlenie jednotlivých etáp. Popis manažmentu rizík sa na tomto mieste sústreďuje najmä na rizikový posudok, ktorý sa skladá z troch etáp: identifikácia rizík, analýza rizík a vyhodnotenie rizík. V rámci analýzy rizík je popísaných niekoľko zaužívaných a úspešných metód, ktoré využívajú organizácie na celom svete. Táto diplomová práca sa zameriava hlavne na Markovovu analýzu a analýzu pomocou Petriho sietí, nakoľko sú tieto dve metódy v kontexte manažmentu projektov pomerne nové a zatiaľ nie veľmi rozšírené.

Štvrtá kapitola je vyhradená pre analýzu a návrh aplikácie, ktorá bude slúžiť ako podpora analýzy rizík zvolenou metódou. Z uvedených dvoch metód dostala prednosť Markovova analýza, čomu napokon zodpovedá aj príslušný návrh aplikácie.

Ďalšia kapitola sa už venuje samotnej implementácii a jej výsledkom. Funkcie programu sú podrobne popísané a podporené radou snímok samotnej aplikácie. Záverom kapitoly je predstavených niekoľko modelov, ktoré prakticky ozrejmujú Markovovu analýzu v prostredí implementovanej aplikácie a sú prínosom pre manažment rizík IT projektov i projektov všeobecne.

Záverom práce sú hodnotené a diskutované dosiahnuté výsledky spolu s návrhmi možného rozšírenia v budúcnosti.

Cieľom tejto diplomovej práce je teda navrhnutie a implementácia softvérového systému, ktorý zdokonalí súčasný manažment rizík a predstaví v ňom pomerne málo zaužívanú Markovovu analýzu. Táto práca sa snaží o demonštrovanie úspešnosti a účelnosti tejto metódy, ktorá je obvykle spájaná len s modelovaním rôznych technických systémov, ale v manažmente rizík projektov ako celkov je pomerne neznáma.

Práca priamo naväzuje na Semestrálny projekt, ktorý obsahoval časť popisu manažmentu rizík a bolo v ňom tiež načrtnuté smerovanie navrhnutého softvéru. V texte sa vyskytuje množstvo odkazov na použitú literatúru, pričom informácie z rôznych zdrojov boli kombinované tak, aby zodpovedali forme diplomovej práce.

Kapitola 2

Manažment projektov

Táto diplomová práca pojednáva (ako vyplýva z jej názvu) o analýze rizík v manažmente projektov. Skôr ako sa dostaneme k samotnej analýze rizík a k jej metódam, je nevyhnutná aspoň základná znalosť o manažmente projektov, čo je témou tejto kapitoly. Okrem samotného definovania pojmu manažment projektov si tiež ozrejmime termín „projektový manažment“ a rozdiel medzi týmito (na prvý pohľad totožnými) pojmami.

2.1 Projekt

Nosným termínom manažmentu projektov je samozrejme projekt. Jedna z mnohých podobných definícií predstavuje projekt ako proces plánovania a riadenia rozsiahlych operácií. Veľká časť laickej verejnosti si pod týmto pojmom predstavuje dokumentáciu k určitým procesom a postupom, čo samozrejme nie je úplne správne. Projekt je predovšetkým presný postup určitých činností s jasne a konkrétne stanovenými cieľmi. Na dosiahnutie uvedených cieľov je definovaná stratégia a tiež zdroje a náklady, ktorými organizácia disponuje. Podľa prof. Ing. Štefana Kimličku, PhD. možno všeobecné charakteristiky projektu popísať nasledovne [16]:

- projekt má jasný začiatok a koniec, je jedinečný a systémový
- projekt je charakterizovaný obmedzenými zdrojmi, neistotou a **rizikom** (viac v kapitole 3.3.4)
- v rámci uvedených obmedzených zdrojov spolu s limitovanými nákladmi a časom je projekt snaha o dosiahnutie zmeny a požadovanej kvality
- projekt je *jedinečný a unikátny súbor činností*. Tie sa od rutinných činností odlišujú ako obsahom, tak aj cieľom
- z jasne definovaného začiatku a konca je zřejmé, že projekt má dočasný charakter. Z toho vyplýva, že vymedzené zdroje (materiálne, finančné i ľudské) sú k dispozícii práve pre určitý projekt a po jeho ukončení sa zrušia
- v rámci trvania projektu možno definovať množstvo etáp, v rámci ktorých dochádza k zmenám úloh, organizácie a zdrojov
- všetky zdroje projektu sú organizované na dosiahnutie vytýčeného cieľa

- komplexné projekty vyžadujú vedomosti a schopnosti odborníkov rôznych oblastí
- *projekt nie je* periodicky sa opakujúca činnosť a ani jednoduchá úloha (napr. obstarania zariadení a pod.)

V dnešnej dobe už poznáme množstvo teoretických postupov a metodológií, pomocou ktorých možno dosiahnuť úspešný koniec projektu. Medzi hlavné faktory, ktoré ovplyvňujú úspešnosť projektu patria najmä skúsenosti, schopnosti a vedomosti projektového manažéra, jasne vymedzené obchodné ciele a podpora zvyšku manažmentu organizácie. Ďalším kľúčovým faktorom je spätná väzba zákazníka, resp. budúceho užívateľa, teda jeho zapojenie do procesov projektu. Na úspechu projektu sa v neposlednom rade podieľa tiež presne definovaný rozpočet, projektová metodika a spoľahlivé odhady. Úspešne ukončený projekt musí obvykle spĺňať nasledovné body [7]:

- ukončenie projektu nastalo v stanovenom termíne
- neboli prekročené vymedzené finančné zdroje (stanovený rozpočet)
- bol naplnený rozsah a požadovaná kvalita projektu (projekt bol akceptovaný zákazníkom)
- ukončenie projektu nenarušilo fungovanie organizácie, pod záštitou ktorej projekt prebiehal a prácu v nej
- množstvo vykonaných zmien v projekte, ktoré vybočujú od zadania projektu bolo minimálne a uvedené zmeny boli odsúhlasené

Nesplnenie jedného alebo viacerých predošlých bodov môže viesť (a v praxi často vedie) k neúspešnému ukončeniu projektu.

2.2 Projektový manažment

Vzťah pojmov *projektový manažment* a *manažment projektu* je podobný ako v prípade pojmov ekonómia a ekonomika. Kým ekonómia je veda, ktorá formuluje ekonomické zákony a skúma najvšeobecnejšie vzťahy medzi ekonomickými subjektami, ekonomika pojednáva priamo o subjekte alebo prostredí, kde je vyvíjaná ekonomická činnosť.

Analogicky môžeme teda **projektový manažment** definovať ako vedu, resp. filozofiu prístupu k riadeniu projektu s relatívne krátkodobou trvácnosťou, kde je jasne stanovený súbor cieľov a definovaný počiatok a koniec projektu a taktiež zdroje, náklady a požadovaná kvalita. Krátkodobá trvácnosť je v tomto kontexte dosť relatívny pojem, ktorý sa odvíja od typu zamerania projektu, resp. jeho odvetvia. V prípade projektu stavby domu teda možno hovoriť o troch mesiacoch, pričom napr. v prípade stavby nemocnice o dvoch rokoch.

Mnoho manažérov nikdy nenadobudlo dostatok skúseností a vedomostí o projektovom manažmente, iba tie, ktoré získali priamo praxou. Tým pádom sú chyby neustále opakované a koncept, ktorý sa mal stať kľúčový pri riešení neefektívneho manažmentu sa sám pretransformuje na neefektívny manažment. Preto je veľmi dôležité hlavne u začínajúcich manažérov, aby si koncept všeobecného projektového manažmentu dostatočne osvojili, k čomu dnes môže dopomôcť viacero publikácií [19].

2.3 Manažment projektov všeobecne

V predošlej kapitole sme definovali projektový manažment ako abstraktnejší pojem, resp. ako vedou alebo filozofiou prístupu k projektom vo všeobecnosti. Naproti tomu **manažment projektov** je už konkrétnejší prístup, ktorý vyžaduje špeciálne metódy, techniky a nástroje riadenia. Súhrnným názvom pre uvedené atribúty je metodika projektu, ktorú možno rozčleniť na tieto základné procesy [10]:

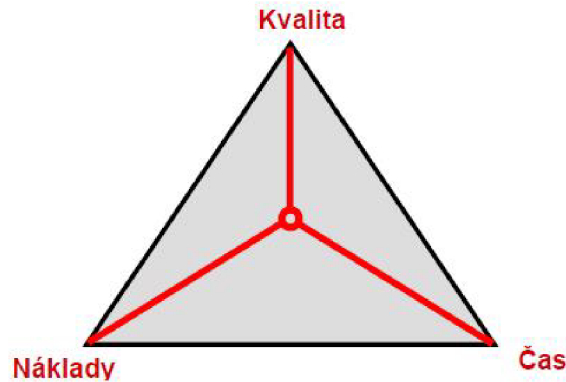
- *inicializácia* – v tejto etape okrem iného vyberáme manažéra projektu, čo je veľmi dôležitý moment. Spravidla sa musí jednať o osobu s kvalitnými komunikačnými a organizačnými schopnosťami, pričom veľkou výhodou sú tiež technické vedomosti
- *plánovanie projektu* – jedná sa o popis toho, čo chceme aby sa stalo. Takýto popis pomáha zlepšiť vykonávanie projektu a presne určiť jeho ciele. Plán teda býva základným podkladom pre riadenie práce, z čoho je zrejmé, že sa počas celého trvania projektu kladie veľký dôraz na jeho dodržiavanie. Dôležitým abstraktným elementom v tejto fáze je čo najpresnejší odhad. Ten je nevyhnutný pre definovanie požiadaviek na náklady a zdroje a tiež pre ich rozčlenenie do jednotlivých činností, pričom nemožno zabúdať aj na správny odhad trvania týchto činností
- *riadenie realizácie projektu* – je to proces, ktorý smeruje k splneniu definovaného plánu. Vykonávajú sa tu preventívne činnosti predchádzajúce problémom. Vstupnými atribútami tejto etapy sú projektový plán a priebežné správy o výsledkoch práce. Tie sa potom pretransformujú do správ o výkone, čím kontrolujeme celkový vývoj projektu. Kľúčovým procesom pri riadení je tiež riadenie zmien, riadenie tímu a riadenie rizík.
- *vykonávanie projektu* – ide v podstate o postupné vykonávanie činností na základe rozvrhu, resp. plánu
- *ukončenie projektu* – vo fáze ukončenia je dôležité zhrnúť nadobudnuté poznatky a skúsenosti a diskutovať prípadné vzniknuté chyby, ktoré je nutné eliminovať v prípadných ďalších projektoch. Podmienky úspešného ukončenia projektu boli popísané v kapitole 2.1.

Pri uvedenom procese plánovania projektu sa často spomína pojem trojimperatív projektu. Často je znázorňovaný ako trojuholník, kde každá hrana predstavuje jednu dimenziu projektu: čas, náklady a kvalita (obrázok 2.1). Akákoľvek zmena jedného parametru sa premietne aj do ďalších dvoch parametrov, v závislosti od požadovaného dôrazu na daný atribút (v niektorých projektoch je pre zadávateľa dôležitejšia kvalita a čas ako náklady, a pod.) [22].

2.4 Manažment IT projektov

Prvky projektového manažmentu, ktorý je stručne načrtnutý v kapitole 2.2 sú v dnešnej dobe mimoriadne dôležité pri vývoji softvéru, či už sa jedná o organizácie, ktoré spoliehajú na externé vývojárske firmy alebo o organizácie, ktoré si softvér produkujú samy.

Autori Evanco a Verner vo svojej práci [20] zverejnili svoj výskum ohľadom IT produktov, konkrétne čo vedie k ich úspešnému, resp. neúspešnému ukončeniu. Poukazujú teda na fakty, ktoré majú priamy súvis s úspechom projektu a na najčastejšie chyby. Analýza



Obrázek 2.1: Projektový trojimperatív (prevzaté z [22])

prebehla vo viacerých amerických firmách rôzneho zamerania (banky, farmaceutické spoločnosti, poisťovne, atď.), ktoré vyvíjali softvér pre svoje potreby. Celkovo bolo analyzovaných 122 projektov, pričom 62% skončilo úspešne a 38% neúspešne. Tieto čísla môžeme považovať za relevantné v kontexte akéhokoľvek IT projektu, nakoľko skúmané projekty mali najrôznejšie atribúty. Tým sa myslí najmä rôzny počet zamestnancov, ktorí pracovali na projekte, alebo napr. rôzne modely životného cyklu (vodopád, prototypovanie, atď.).

Za záver tohto výskumu možno považovať zistenie, že osoba projektového manažéra je nadmieru dôležitá na dosiahnutie úspešného konca projektu. Dôležitým faktorom neúspechu projektov býva často slabá alebo dokonca žiadna analýza rizík. Tá je hlavnou témou tejto diplomovej práce.

2.5 Zainteresované subjekty

V nasledujúcom texte sa často stretneme s pojmom *zainteresované subjekty* (z angl. *stakeholders*). Slovenský jazyk nemá presný ekvivalent k tomuto anglickému pojmu, ale ako najvýstižnejšia interpretácia sa javí práve termín *zainteresované subjekty*. Môžeme teda hovoriť o jednotlivcoch, skupinách ľudí alebo dokonca o celých organizáciách, ktoré sa nejakým spôsobom zaujímajú o určitý projekt alebo majú na projekt nejaký dopad, prípadne sú projektom samy ovplyvnené.

Identifikácia zainteresovaných subjektov je teda tiež dôležitým aspektom manažmentu projektov, pretože je vždy nutné definovať všetky, resp. čo najviac externých a interných prvkov, ktoré projekt ovplyvňuje a ktorými je ovplyvňovaný. Správna identifikácia zainteresovaných subjektov je dôležitá aj v etape analýzy rizík.

2.6 Analýza rizík v manažmente projektov

V predošlých kapitolách boli stručne načrtnuté pojmy ako *projekt*, *projektový manažment* a *manažment projektu*. Každý z týchto termínov by si zaslúžil oveľa väčší priestor na komplexný popis a pochopenie všetkých aspektov uvedených pojmov, avšak táto diplomová práca len stručne ozrejmuje nevyhnutné základy, ktoré sú potrebné na pochopenie hlavnej témy – analýzy rizík, ktorej budú venované nasledujúce kapitoly.

V dnešnom enormne dynamickom svete, kde je veľká konkurencia, narastá čoraz viac potreba zvládnutia projektového manažmentu a správnej identifikácie rizík projektu. Identifikácia rizík je však len prvým krokom k úspechu, pretože definovanie samotných rizík nevedie k vedomosti o ich možnom dopade na projekt, prípadne celú organizáciu. Tu prichádza na scénu práve spomínaná analýza rizík.

Kapitola 3

Manažment rizík

Už v predošlých kapitolách bolo párkrát naznačené, že spoločnosti a organizácie všetkých typov a veľkostí čelia spravidla pomerne širokému spektru rizík. Tieto riziká potom v rôznej miere ovplyvňujú firemné procesy a predovšetkým ciele, o ktoré firma usiluje. Uvedené ciele možno spájať s radou aktivít v rámci organizácie, počnúc strategickými podnetmi, operáciami, procesmi a projektami. Ďalej nemožno nespomenúť dopady spoločenské, ekologické, technologické, bezpečnostné, marketingové, finančné a iné. [13].

Táto kapitola sa pomerne obsírne venuje jednotlivým etapám manažmentu rizík, s dôrazom na analýzu rizík. Tento proces obvykle chápeme ako etapu manažmentu projektu.

3.1 Riziko a príležitosť

Pri riešení projektu v akomkoľvek odvetví uvažujeme množstvo faktorov, o ktorých máme pochybnosti. Pochopenie týchto faktorov spolu s ich analýzou sú kľúčom k správne rozhodnutiu v ďalšom smerovaní.

Medzi tieto faktory patria náhodné udalosti, ktoré môžu ale nemusia nastať, pričom v prípade ich výskytu majú na celkový projekt nejaký dopad. Práve v závislosti od tohto dopadu rozlišujeme dva typy týchto udalostí [21]:

- **Riziko** je náhodná udalosť, ktorá môže nastať a ak by nastala, mala by *negatívny dopad* na ciele organizácie, resp. projektu. Riziko je teda definované tromi atribútmi: scenárom, pravdepodobnosťou jeho výskytu a mierou dopadu ak by skutočne nastalo.
- **Príležitosť** je tiež náhodná udalosť, ktorá by však v prípade výskytu mala *pozitívny dopad* na ciele organizácie. Príležitosť je definovaná tromi rovnakými atribútmi ako riziko.

Riziko a príležitosť môžu byť považované ako protíahle strany jednej mince. Obyčajne je najjednoduchšie považovať možnú udalosť ako riziko, ak by táto udalosť mala negatívny dopad a pravdepodobnosť výskytu menšiu ako 50%. V prípade, že pravdepodobnosť výskytu rizika presahuje 50%, je možné túto udalosť v pláne projektu uvažovať ako príležitosť, ak by udalosť nenastala [21].

3.2 Štruktúra manažmentu rizík

Úvodom tohto odseku si definujeme dva dôležité pojmy:

Definícia 3.2.1. *Manažment rizík* je súhrn koordinovaných činností, na smerovanie a riadenie organizácie, kde sa berie ohľad na *riziká 3.3.4* [14].

Definícia 3.2.2. *Štruktúra manažmentu rizík* je sada komponentov na zabezpečenie podkladov a organizačných plánov na návrh, realizáciu, kontrolu, zhodnotenie a ustavičné vylepšovanie *manažmentu rizík 3.2.1* v rámci celej organizácie [14].

Štruktúra manažmentu rizík teda poskytuje postupy, činnosti a organizačné plány, ktoré uvažujú a vkladajú manažment rizík do všetkých úrovní v organizácií. Súčasťou tejto štruktúry sú tiež postupy a stratégie, ktoré rozhodujú kedy a ako posúdiť riziko. Zrealizované posudky rizík musia byť obzvlášť jasné v nasledujúcich bodoch [13]:

- súvislosti v organizácií a ciele organizácie
- rozsah a typ rizika, ktorý je možné tolerovať a spôsob ako sa zaoberať rizikami, ktoré sú neakceptovateľné
- vplyv posudku rizík na procesy v organizácií
- metódy a techniky posudku rizík a ich prínos pre proces manažmentu rizík
- zodpovednosť, spoľahlivosť a právomoc pri vykonávaní posudku rizík
- zdroje, ktoré sú k dispozícii na vypracovanie posudku
- spôsob, akým bude posudok oznámený a zhodnotený

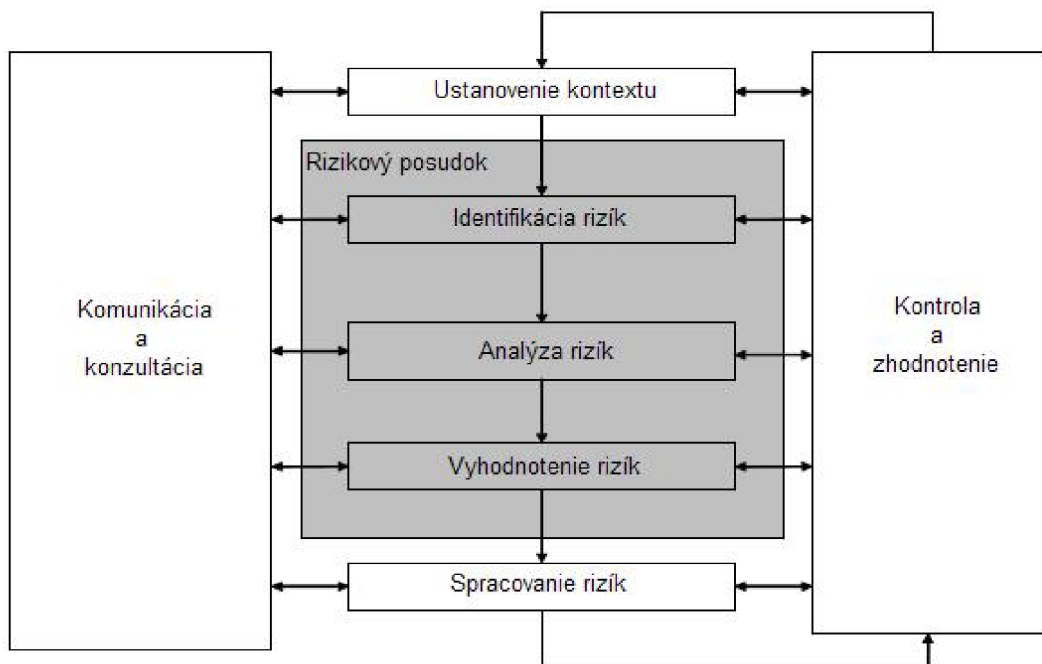
3.3 Rizikový posudok a jeho miesto v manažmente rizík

Rizikový posudok je zahrnutý do nasledujúcich základných stavebných kameňov manažmentu rizík, ktorých normované definície sa dozvieme ďalej v príslušných kapitolách [13]:

- komunikácia a konzultácia
- ustanovenie kontextu
- rizikový posudok (obsahuje identifikáciu, analýzu a vyhodnotenie rizík)
- spracovanie rizík
- kontrola a zhodnotenie

V tejto práci sa najväčší dôraz kladie na tretí bod manažmentu rizík, teda na rizikový posudok. Práve ten zahŕňa pre nás kľúčový pojem – analýzu rizík.

Posudok rizík sa nikdy nevyvíja osamotene. Jedná sa o proces, ktorý je vždy plne integrovaný do ostatných komponent manažmentu rizík. Toto celkové prepojenie znázorňuje obrázok 3.1 [13]:



Obrázok 3.1: Rizikový posudok v rámci managementu rizík (prevzaté z [13])

3.3.1 Komunikácia a konzultácia

Definícia 3.3.1. *Komunikácia a konzultácia* je súvislá a iteratívna činnosť, ktorá sa vykonáva v rámci organizácie s úmyslom poskytnúť, získať alebo zdieľať informácie a tiež s úmyslom zapojiť sa do dialógu s osobou, zodpovednou za kľúčové rozhodnutia s ohľadom na možné riziká [14].

Na obrázku 3.1 možno vidieť, že proces komunikácie a konzultácie je sústavný a prebieha súčasne s jednotlivými etapami posudku rizík. Základná otázka, ktorú si kladieme v tejto etape je: „Kto má byť zapojený do procesu?“ [6]. V tomto smere sa nemyslí výlučne osoba, ktorá je zodpovedná za ďalšie rozhodnutia a smerovanie organizácie, ale tiež osoba, ktorá môže byť určitým rozhodnutím alebo činnosťou ovplyvnená. Takéto osoby nazývame *zainteresovanými subjektami* (viac v kapitole 2.5) [14].

Zahrnutie zainteresovaných subjektov do procesu manažmentu rizík je prospešné v nasledujúcich bodoch [13]:

- vývoj komunikačného plánu
- definovanie kontextu
- uistenie sa, že záujmy zainteresovaných osôb sú zohľadnené a porozumené
- zosúladenie rôznych expertíz za účelom identifikácie a analýzy rizík
- uistenie sa, že rôzne pohľady sú vhodne zohľadnené pri vyhodnotení rizík
- uistenie sa, že riziká sú dostatočne identifikované
- zabezpečenie schválenia a podpory plánu opatrení

3.3.2 Ustanovenie kontextu

Ustanovenie kontextu definuje základné parametre pre manažment rizík a vytvára vstupné podmienky pre nasledujúce činnosti v organizácii. Uvažujú sa ako vnútorné, tak aj vonkajšie parametre, ktoré sú relevantné pre organizáciu ako celok a taktiež sa vytvára prvotné pozadie pre posúdenie rizík [13].

Definovanie vonkajšieho kontextu zahrňuje zoznámenie sa s prostredím, v ktorom organizácia operuje. Jasne sú definované faktory kultúrne, politické, legislatívne, regulačné, finančné a ekonomické a to všetko na úrovni medzinárodnej, štátnej, regionálnej i lokálnej [13].

Pod konkrétnym pojmom ustanovenie kontextu sa rozumie najmä vybudovanie organizačného a projektového prostredia, v ktorom sa v neskorších fázach bude vytvárať posudok rizík. V tejto fáze sa tiež špecifikujú hlavné ciele a výstupy, definujú sa kritériá úspechu a prebieha diskusia o následkoch prípadných identifikovaných rizík. Pre účely spomínanej identifikácie rizík sa v tejto etape vytvára akási sada kľúčových elementov na jej štrukturalizáciu a neskoršie posúdenie a zhodnotenie. Poslednými aktivitami tejto fázy sú zhodnotenie organizačnej a projektovej dokumentácie a v neposlednom rade tiež vykonanie analýzy dopadu úspechov a neúspechov projektu na všetky zainteresované subjekty [6].

Vstupnými zložkami etapy ustanovenia kontextu sú kľúčové projektové dokumenty, medzi ktoré patria napr. strategický plán uskutočnenia projektu, potvrdenia a oprávnenia ohľadom projektu, odhad nákladov, časového plánu a poľa pôsobnosti. Ďalej netreba zabudnúť ani na technologické návrhy a štúdie, ekonomické analýzy a ostatné relevantné dokumenty o projekte a jeho účele. Hlavnou výstupnou zložkou etapy je stručný prehľad cieľov projektu a organizácie a tiež prehľad špecifických kritérií pre úspech. Ďalej je to samozrejme súhrn cieľov a oblastí pre posudok rizík a spomínaná sada kľúčových elementov, potrebná pre identifikáciu rizík [6].

3.3.3 Rizikový posudok

Posúdenie rizík je komplexný proces, ktorý pod sebou zastrešuje tri väčšie celky: identifikáciu, analýzu a vyhodnotenie rizík (viz obr. 3.1). Týmto trom podetapám sa budeme obšírnejšie venovať v samostatných kapitolách.

Riziká môžu byť posudzované a odhadované na rôznych úrovniach: na globálnej úrovni celej organizácie, na úrovniach jednotlivých oddelení a samozrejme v elementárnej rovine jednotlivých aktivít a procesov. Pre rôzne úrovne a kontexty sú vhodné rôzne techniky, nástroje a metódy posudzovania. Celkovo nám posúdenie rizík poskytuje predovšetkým porozumenie rizík a ich príčin a tiež ich následkov a pravdepodobností výskytu. To slúži ako kľúčový vstupný atribút pre rozhodovanie v nasledujúcich bodoch [13]:

- má vôbec význam zaoberať sa danou aktivitou?
- ako maximalizovať príležitosti, resp. pozitívne dopady?
- je výhodné zavádzať opatrenia proti danému riziku?
- ako si správne vybrať medzi viacerými scenármi pri rôznych rizikách?
- opatrenia proti ktorým rizikám majú najvyššiu prioritu?
- ktorá stratégia zavádzaní opatrení proti rizikám zníži ich negatívny dopad na minimum?

Hlavným účelom posudku rizík je poskytnutie informácií a analýz na základe predošlých poznatkov a dôkazov. Posudok sa potom zohľadňuje pri ďalšom smerovaní a rozhodovaní ako sa postaviť k možnému riziku a ako si zvoliť medzi viacerými možnosťami a scenármi. Je teda zrejmé, že posúdenie rizík je pre všetky spoločnosti a organizácie veľmi dôležité a plynie z neho množstvo úžitkov. Prvým pozitívom je bezpochyby porozumenie rizika a jeho dopadu na ciele organizácie, resp. projektu, čím sa zvyšuje informovanosť osôb, ktoré sú kompetentné v ďalšom rozhodovaní a smerovaní (zainteresované subjekty). To okrem iného znamená predovšetkým výber najvhodnejšej varianty eliminácie rizika a objavenie, resp. definovanie slabých článkov organizácie. Posúdenie rizík takto tiež prispieva k prevencii pred rôznymi incidentmi, nehodami a katastrofickými scenármi. Zaujímavým prínosom je tiež vyhodnotenie rizika v zmysle, či je vôbec nutné a výhodné zabezpečiť príslušné opatrenia ak rizikový scenár nastane, alebo je nejakým spôsobom výhodnejšie ochranné opatrenia nezavádzať. Jedná sa obvykle o prípady, keď sa nejaké riziko definuje, ale náklady na jeho odstránenie by boli príliš vysoké, resp. jeho celkový (negatívny) dopad je zanedbateľný [13].

3.3.4 Spracovanie rizík

Účel etapy spracovania rizík je dopracovanie sa k rozhodnutiu, ako zareagujeme na jednotlivé identifikované a rozanalyzované riziká, aby sme minimalizovali ich negatívny efekt. Je zrejmé, že ak sa neurčí vhodná reakcia, resp. opatrenia, celý proces posudku rizík (identifikácia, analýza, vyhodnotenie) bol zbytočný. Vo fáze spracovania rizík sa teda vlastne konvertujú závery analýzy a vyhodnotenia na konkrétne kroky redukcie dopadov rizík. Hlavnými vstupnými atribútmi etapy sú okrem zoznamu rizík (spolu s ich prioritami a pravdepodobnosťami) aj dokumenty, pojednávajúce o plánoch organizácie a jej rozpočte. Na základe týchto vstupov sa potom definujú možnosti a scenáre opatrení, ktoré znižujú pravdepodobnosť výskytu rizika, alebo v prípade jeho objavenia minimalizujú jeho negatívne následky. V prípade výskytu príležitostí (kapitola) sa zase volia scenáre, ktoré sa najviac snažia vyťažiť z ich pozitívneho efektu [6].

Vhodných reakcií na korektne identifikované a vyhodnotené riziká je mnoho, ale všetky obvykle spadajú do nasledovných kategórií [21]:

- **akceptácia (ničnerobenie)**

S výskytom rizikovej situácie sa síce ráta, ale neplánujú sa žiadne opatrenia na elimináciu alebo zmiernenie dopadov. Takýto scenár je vhodný pri rizikách, kde by náklady na prevenciu presiahli náklady odstránenia negatívnych dopadov. Jedná sa teda o riziká s nízkonákladovým dopadom a malou pravdepodobnosťou výskytu.

- **zvýšenie rizika**

Analýza rizík občas odhalí, že organizácia momentálne vynakladá značné prostriedky na zvládnutie určitého typu rizika, ktoré sú neúmerne veľké v porovnaní s úrovňou ochrany, ktorá je takto zabezpečená. V takýchto prípadoch sa väčšinou javí logické zredukovať túto úroveň ochrany a presmerovať zdroje na prevenciu proti iným rizikám. Tým sa zabezpečí výkonnejšia ochrana a prevencia proti celkovej skupine rizík, ktorým organizácia čelí. Príkladom z praxe môže byť napríklad odstránenie enormne nákladných bezpečnostných regulácií pre jadrové elektrárne, ktoré eliminujú v podstate bezvýznamné riziká. Ďalším scenárom z praxe je napríklad zrušenie legislatívnej požiadavky testovať všetky zabitú dobytku na BSE a použitie takto ušetrených prostriedkov na nákup efektívnejších zariadení a liekov v nemocniciach.

Takéto scenáre, resp. reakcie na riziká sú zväčša logické, avšak častokrát politicky a spoločensky neakceptovateľné. Existuje len málo politikov a riaditeľov, ktorí nemajú problém prezentovať verejnosti, že práve odsúhlasili zníženie prevencie nejakého rizika.

- **získanie väčšieho množstva informácií**

Počas procesu analýzy rizík sa v kontexte nejakého rizika objaví príliš veľká miera nejakej neistoty, ktorá v etape spracovania rizík spôsobuje markantný problém s rozhodnutím, ako sa k riziku postaviť (v tomto prípade pod pojmom „neistota“ nemyslíme prirodzenú náhodnosť výskytu). Táto neistota môže byť častokrát zredukovaná prostým dodatočným získaním väčšieho množstva informácií. Nové poznatky sú potom vstupom pre opätovne vykonanú analýzu rizík, ktorá usmerní kompetentnú osobu k lepšiemu rozhodnutiu.

- **zrušenie rizika (eliminácia)**

Do tohto typu reakcie na riziko zahrňujeme zmenu výrobných metód, aktivít, projektového plánu, investičných stratégií, atď. takým spôsobom, že identifikované riziko viac nie je relevantné. Scenár eliminácie sa zvyčajne zavádza pri rizikách s markantným negatívnym dopadom a s vysokou pravdepodobnosťou výskytu. Všeobecné príklady z praxe sú napr.:

- použitie osvedčených a otestovaných technológií namiesto vlastných nových, ktoré boli pôvodne uvažované
- zmeniť krajinu, kde má byť postavená továreň, za účelom vyhnutia sa politickej nestability

V tomto prípade je nutné si uvedomiť, že takouto radikálnou zmenou plánov vzniká reálna šanca výskytu nových (a často omnoho závažnejších) rizík.

- **redukcia (zmiernenie dopadu)**

Pod redukciami možno v tomto prípade chápať celú škálu techník, ktoré sa používajú na zníženie pravdepodobnosti výskytu rizika alebo na zmiernenie jeho dopadu alebo oboje. Klasické všeobecné príklady sú:

- tvoriť nadbytočne (záložné počítače, náhradné diely)
- vykonávať viacero kvalitatívnych testov a inšpekcií
- zabezpečiť kvalitnejšie školenie personálu

Redukčné stratégie sú používané takmer pre všetky typy a úrovne rizík, pričom zredukované riziko nesmie vykazovať vysokú „bolestnosť“ (dopad a pravdepodobnosť výskytu). Okrem toho musia ušetrené náklady, ktoré by bolo nutné spotrebovať pri odstránení dopadov rizika pred redukciami, prevyšovať náklady na samotné redukčné opatrenia.

- **predvídanie výnimočných situácií**

V tomto prípade sa jedná o plánovanie akejkoľvek optimalizácie reakcií na riziká, ktoré majú vypuknúť. Tieto plány obvykle vznikajú v spojení dvoch spomenutých stratégií: redukcie a akceptácie. Plán na predvídanie výnimočných udalostí jasne definuje jednotlivcov prípadne mechanizmy, ktoré preberajú zodpovednosť za monitorovanie

výskytu rizika a jeho následnú redukciu. Jednoducho povedané, definuje sa čo robiť, kto by to mal urobiť, v akom poradí, s akou prioritou, atď. Typické príklady z praxe sú nasledovné:

- mať na mieste pripravený vycvičený tím požiarnikov
- mať pripravenú tlačovú konferenciu
- disponovať zoznamom telefónnych čísel a emailov osôb, ktoré nutno urýchlene kontaktovať v prípade výskytu rizika
- pripevniť dostatok záchranných člnov na zaoceánsku výletnú loď

• **zabezpečenie rezervných zdrojov**

Zabezpečenie rezervných zdrojov nastáva v rámci organizácie výlučne na základe rozhodnutia manažmentu. Jedná sa o alokovanie určitého typu zdrojov v prípade rizík, ktoré môžu nastať. Stratégia je vhodná v prípade výskytu rizík malého až stredného dopadu. Príklady sú:

- vyhradenie väčšieho množstva finančných prostriedkov pre prebiehajúci projekt
- zabezpečenie extra času pre dokončenie projektu
- disponovanie finančnými rezervami
- disponovanie väčším množstvom tovaru na sklade predajní počas štátnych sviatkov
- hromadenie zásob jedla a liekov

• **poistenie**

V podstate sa jedná o typ stratégie z kategórie „redukcia“, ale je natoľko rozšírený, že je vhodné uviesť ho zvlášť. Pokiaľ poisťovacia spoločnosť správne určila všetky kalkulácie, platia jej klienti v dlhodobom horizonte o niečo vyššiu čiastku, ako by zaplatili jednorázovo pri výskyte rizika. Vo všeobecnosti sa teda dá povedať, že poisťujeme riziká, ktoré majú pre nás vyššiu cenu, ako je ich objektívna hodnota. Existuje však ešte alternatívny výklad, keď si uvedomujeme oveľa vyššie vystavenie riziku ako je tomu u obvyčajného klienta. V takom prípade môže príjem z poistky výrazne prevýšiť škody a razom sa táto stratégia stáva vysoko atraktívnou.

• **prenesenie rizika**

Táto metóda je akousi manipuláciou, počas ktorej sa riziko (obvykle len jeho časť) prenáša z jednej zainteresovanej osoby na druhú. Obvyklá je metóda prenosu rizika prostredníctvom kontraktov, kde sa nejakým spôsobom pokutuje nedostatočný výkon dodávateľa. Jedná sa o častú metódu, avšak mnohokrát pomerne neefektívnu. Typickými príkladmi sú:

- zmluvná doložka o pokute pri nedodržaní termínu projektu
- garancia kvality produktu
- prenájom nehnuteľnosti namiesto jej kúpy
- objednanie si reklamnej kampane od reklamnej agentúry, pričom splátky budú podmienené dohodnutou mierou úspechu výrobku alebo služby

Niekedy je dokonca vhodné prebrať určité riziko na seba v prípade, že nastáva pre nás výhoda odbremenenia druhej strany. Väčšinou sa jedná o prípady, keď nám určitá aktivita s malým preneseným rizikom prináša oveľa väčší úžitok ako druhej zainteresovanej strane.

- **zrušenie projektu**

Pri rizikách s obzvlášť závažným dopadom a výskytom pravdepodobnosti je nutné uvažovať i variantu zrušenia projektu. Takýto scenár obvykle nastáva ešte pred samotným začiatkom práce na projekte, prípadne v jeho rannej fáze rozpracovanosti.

3.3.5 Kontrola a zhodnotenie

Podobne ako v prípade etapy „komunikácia a konzultácia“ (kapitola 3.3.1), aj v tomto prípade sa jedná o súvislú a iteratívnu činnosť, ktorá svojím spôsobom zasahuje do ostatných zložiek manažmentu rizík a manažmentu projektu ako takého. Sústavná kontrola a zhodnotenie je dôležitá časť implementácie projektu, hlavne pri väčších projektoch a projektoch v dynamickom prostredí. Na základe tejto etapy je zaistená neustála detekcia a spracovanie nových rizík a teda plánovanie a implementačné činnosti postupujú efektívnejšie. Výstupom etapy je akýsi zoznam sledovaných rizík, ktorého neustále sledovanie je v kompetencii vedúceho projektu. Pri väčších projektoch existuje takýchto zoznamov niekoľko, v závislosti od počtu oddelení organizácie a ich manažérov. Týmto sa rozdeľujú riziká do niekoľkých polí pôsobností a viac či menej rovnomerne sa rozkladá zodpovednosť [6].

Pre názornosť môžeme definovať niekoľko bodov, ktoré sa sústavnou kontrolou a hodnotením verifikujú [13]:

- správnosť predpokladu určitého rizika
- správnosť predpokladov, na ktorých je založené posúdenie rizík (včetně vonkajšieho a vnútorného kontextu)
- výsledok analýzy a hodnotenia rizika sa zhoduje s očakávaniami
- výsledky posudku rizík sú v zhode s výsledkami predošlých skúseností
- správnosť aplikovania techník posudku rizík
- spracovanie rizika na základe zvolenej stratégie je efektívne

3.4 Proces posúdenia rizík

Táto kapitola rozširuje stručné zhrnutie procesu posudku rizík v kapitole 3.3.3 a jeho troch hlavných zložiek: identifikácie, analýzy a vyhodnotenia rizík. Objasneniu princípov niekoľkých modelovacích techník podetapy „analýza rizík“ a ich vzájomnému porovnaniu bude venovaná opäť špeciálna kapitola 3.5.

Posudok rizík poskytuje manažérom a ostatným osobám, ktoré sú zodpovedné za kľúčové rozhodnutia spresnený pohľad na porozumenie určitých rizík, dôležitých pri úspešnom a efektívnom dosiahnutí cieľov. Výstup tejto etapy je teda vstup pre rozhodovacie procesy v celej organizácii, resp. v manažmente projektu. Je zrejmé, že posudok rizík vyžaduje multidisciplinárny prístup, nakoľko zastrešuje široké pole pôsobností a prípadných konsekvencií [13].

3.4.1 Identifikácia rizík

Definícia 3.4.1. *Identifikácia rizík* je proces hľadania, rozoznávania a popisovania *rizík* 3.3.4 [14].

Účelom tejto etapy je identifikovanie, čo všetko sa môže stať a aké situácie môžu nastať, ktoré nejakým spôsobom ovplyvnia aktivity a ciele projektu alebo organizácie. V momente keď je riziko identifikované, je povinnosťou organizácie definovať všetky kontrolné orgány v tomto kontexte, akými sú návrhové rysy, zainteresované subjekty, procesy a systémy. Proces identifikácie rizík v sebe zahŕňa definovanie príčin a zdrojov rizík, udalostí, situácií a okolností, ktoré majú nejaký materiálny dopad na ciele organizácie a tiež definovanie typu tohto dopadu [13].

Proces identifikácie rizík musí byť zrozumiteľný, pretože riziká, ktoré neboli identifikované, nemôžu byť posúdené. Ich neskorší výskyt potom môže logicky ohroziť úspech projektu a zapríčiniť nepríjemné prekvapenia. Je teda nevyhnutné, aby bol tento proces vhodne štrukturovaný, používajúc kľúčové elementy na systematické skúmanie rizík v každej oblasti a poli pôsobnosti v rámci projektu. Vstupnými zložkami identifikácie rizík sú obvykle historické dáta, teoretické analýzy, empirické poznatky a analýzy, relevantné názory členov tímu a expertov v danom obore a samozrejme tiež „obavy“ zainteresovaných subjektov. Výstupom etapy je samozrejme zrozumiteľný zoznam možných rizík a ich popis [6].

Na identifikáciu rizík existuje niekoľko nástrojov a techník. Najbežnejšie z nich sú nasledovné [6]:

- riadená diskusia o riešení nových problémov (brainstorming)
- skúmanie lokálnych i externých skúseností s podobnými aktivitami a projektami, vrátane analýzy správ o ukončení takých projektov a ich auditov
- kontrolné zoznamy
- rozhovory a úzko zamerané skupinové diskusie
- analýza možného scenára
- prieskumy a dotazníky

Bez ohľadu na zvolenú techniku je vždy nutné zahrnúť do identifikácie aj ľudské a organizačné faktory, nie len udalosti a scenáre spojené s „hardvérom“ a „softvérom“ [13].

P – I tabuľky

Ako už bolo spomenuté, fáza identifikácie rizík sa snaží o definovanie rizík, ohrozujúcich ciele projektu alebo organizácie. Je určite dôležité venovať pozornosť všetkým rizikám, avšak nepochybne najmä tým, ktoré sú najväčšou hrozbou. Veľkosť tejto hrozby, resp. *úroveň rizika* (definícia 3.4.2) sa často vyjadruje pomocou tzv. P – I tabuľky. P – I tabuľka umožňuje rýchlu a jednoduchú vizualizáciu dôležitosti všetkých rizík, ktoré sa môžu v projekte objaviť. Zložka tabuľky **P** (od anglického *probability*) predstavuje *pravdepodobnosť* výskytu určitého rizika a zložka **I** (od anglického *impact*) znázorňuje *dopad* rizika [21].

Definícia 3.4.2. *Úroveň rizika* je veľkosť (rozsah) rizika alebo kombinácie rizík, ktorá je vyjadrená kombináciou následkov rizika a pravdepodobnosti rizika [14].

Tvorca posudku rizík musí vhodne popísať pravdepodobnosť a dopad každého rizika zvolením vopred stanovených fráz (napr.: nulový, veľmi nízky, stredný, vysoký). Ku každej fráze je priradený rozsah hodnôt, za účelom udržania vzájomnej nespornosti odhadov jednotlivých rizík [21].

Príklad takýchto rozsahov s priradenými frázami možno vidieť v tabuľke 3.1.

Katagória	Pravdepodobnosť (%)	Zdržanie (dni)	Náklady (tis. Kč)
Veľmi vysoká	10 – 50	> 100	> 1000
Vysoká	5 – 10	30 – 100	300 – 1000
Stredná	2 – 5	10 – 30	100 – 300
Nízka	1 – 2	2 – 10	20 – 100
Veľmi nízka	< 1	< 2	< 20

Tabuľka 3.1: Príklad rozsahu hodnôt, ktorý je asociovaný s pravdepodobnosťami a dopadom na projekt (inšpirované [21])

Rozsahy hodnôt môžu byť definované takým spôsobom, že ich interpretácia sa chápe v kontexte dopadu na celý projekt alebo organizáciu ako celok [21].

Takáto plošná perspektíva, ilustrujúca pôsobenie rizík na celú organizáciu je znázornená tabuľkou 3.2.

Katagória dopadu	Popis
Katastrofický	Ohrozenie existencie celej organizácie
Veľký	Nie je viac možné dosiahnuť stanovené ciele organizácie
Stredný	Zredukovaná možnosť dosiahnutia cieľov organizácie
Malý	Menšie obchodné trhliny, avšak malý efekt na ciele organizácie
Bezvýznamný	Žiadny dopad na strategické ciele

Tabuľka 3.2: Príklad popisov, asociovaných s dopadom rizík na celú organizáciu (inšpirované [21])

Tabuľka 3.3 znázorňuje príklad vizualizácie relatívnej dôležitosti identifikovaných rizík. V tomto fiktívnom príklade bolo v organizácii identifikovaných 12 rizík (číslovanie od 1 do 12), pričom každé riziko je začlenené do bunky P – I tabuľky na základe jeho elementárnych vlastností. Riziká s číselným označením 12, 8, 1 a 11 predstavujú v tomto prípade najväčšie hrozby.

<i>Dopad</i>	Katastrofický		6			12, 8
	Veľký	10			1	11
	Stredný	2		9		
	Malý			3, 4		
	Bezvýznamný	5				7
	Nízka	V. nízka	Stredná	Vysoká	V. Vysoká	
	<i>Pravdepodobnosť</i>					

Tabuľka 3.3: Príklad P – I tabuľky (inšpirované [21])

Skóre P – I tabuľky môže byť použité k ohodnoteniu jednotlivých rizík. Ku každej

kategórií v tabuľke sa priradí hodnota (veľmi nízky = 1, nízky = 2, atď.), ktorá sa neskôr použije pri finálnom ohodnotení úrovne rizika S (definícia 3.4.2). Úroveň rizika sa potom vypočíta vzťahom [21]:

$$S = P + I$$

Príklad výpočtu úrovne rizika je znázornený tabuľkou 3.4. Zelenou farbou sú označené riziká nízkej úrovne, žltou strednej a červenou vysokej úrovne.

Dopad	Katastrofický	6	7	8	9	10
	Veľký	5	6	7	8	9
	Stredný	4	5	6	7	8
	Malý	3	4	5	6	7
	Bezvýznamný	2	3	4	5	6
		Nízka	V. nízka	Stredná	Vysoká	V. Vysoká
		Pravdepodobnosť				

Tabuľka 3.4: Príklad ohodnotenia úrovne rizík (inšpirované [21])

3.4.2 Analýza rizík

Definícia 3.4.3. *Analýza rizík* je proces na porozumenie charakteru *rizika* 3.3.4 a na určenie *úrovne rizika* 3.4.2 [14].

Analýza rizík je teda podľa definície proces komplexného porozumenia rizík. Jej výsledky sú dôležitým faktorom kľúčových rozhodnutí, či určité riziko treba brať do úvahy, resp. ako sa k nemu postaviť a aké stratégie a metódy spracovania použiť [13].

Táto analýza pozostáva najmä z definitívneho určenia prípadných následkov rizík a pravdepodobnosti ich výskytu. Tieto dva najzávažnejšie faktory sa potom kombinujú pri určení finálnej úrovne rizika (kapitola 3.4.1). Etapa analýzy rizík tiež zahŕňa uvažovanie zdrojov a príčin rizík. To v podstate znamená, že je tiež nutné identifikovať faktory, z ktorých vyplynuli dané hodnoty pravdepodobnosti a dopadu. V praxi existuje mnoho metód a modelov analýzy rizík (v praxi najpoužívanejšie boli vymenované už v kapitole o identifikácii rizík - 3.4.1). Ďalšie techniky a modely budú spomenuté v ďalších častiach tejto diplomovej práce, pričom sa zameriame hlavne na Markovovu analýzu a analýzu Petriho sieťami [13].

Analýza rizík obyčajne zahŕňa odhad širšieho rozsahu potenciálnych konsekvencií, ktoré môžu vzniknúť z určitej udalosti, situácie alebo okolnosti a ich príslušnú pravdepodobnosť. Avšak v niektorých prípadoch, zvlášť keď sa budúce následky javia byť pomerne bezvýznamné (alebo je takmer nulová pravdepodobnosť výskytu), sa analýza môže odvíjať aj od jediného parametru, ktorý je pre rozhodnutie v ďalšom smerovaní dostačujúci [13].

Metódy používané v analýze rizík môžu byť *kvalitatívne*, *polokvantitatívne* a *kvantitatívne*. Stupeň požadovanej detailnosti výsledkov závisí od jednotlivej aplikácie danej metódy, od spoľahlivých dát, ktoré sú k dispozícii a od rozhodovacích potrieb organizácie. Niektoré metódy a úroveň ich detailnosti môžu byť občas dokonca predpísané legislatívne [13].

- **Kvalitatívna analýza**

Kvalitatívny posudok definuje následky, pravdepodobnosti a úrovne rizík na základe úrovni dôležitosti (veľmi vysoká, vysoká, stredná, atď.). Dopady a pravdepodobnosti môže kombinovať a vyhodnocuje výslednú úroveň rizika oproti kvalitatívnym kritériám [13].

Kvalitatívny prístup je oproti kvantitatívnemu značne jednoduchší avšak dostačujúci pre mnoho účelov v širokej škále projektov. Jednoduché ohodnotenie pravdepodobnosti a dopadu rizika pomocou P – I tabuľky (kapitola 3.4.1) tiež spadá do kvalitatívnej analýzy. Tento prístup je vo všeobecnosti obzvlášť prospešný v prvotných fázach rôznych ohodnotení a posudkov, keď je nutná rýchla, jednoduchá a efektívna analýza. Málokedy je však tento prístup dostatočný v komplikovanejších prípadoch a v situáciach, kde je nutný presnejší a vierohodnejší posudok [6].

• Polokvantitatívna analýza

Polokvantitatívne metódy používajú numerické hodnotiace škály pre dopady a pravdepodobnosti rizík, ktoré pri zisťovaní celkovej úrovne rizík kombinujú použitie rôznych vzorcov, resp. rovníc. Uvedené škály sú zväčša lineárne alebo logaritmické, pričom používaných vzorcov je tiež celá rada [13].

Na rozdiel od kvalitatívneho prístupu, v tomto prípade sa pravdepodobnosť a dopad neurčujú priamo, ale začína sa tzv. kvalitatívnymi škálami. Tie sa potom transformujú na numerické hodnoty, ktoré sa potom používajú ako indikátory alebo nepriame miery pravdepodobností, dopadov a priorít. Takéto nepriame alebo polokvantitatívne indikátory sú adekvátne v prípade potreby komparatívnych výsledkov. Obzvlášť keď je potrebný rýchly prieskum atribútov projektu, aby management organizácie presne vedel na ktoré časti je nutné sa upriamiť. Cieľom je ohodnotenie rizík od najviac rizikových po najmenej rizikové (absolútne miery nie sú obvykle nutné) [6].

• Kvantitatívna analýza

Kvantitatívna analýza odhaduje praktické hodnoty konsekvencií a pravdepodobností a produkuje hodnoty úrovne rizík v špecifických jednotkách, ktoré sú definované pri vytváraní kontextu. Plne kvantitatívna analýza nemusí byť vždy možná alebo žiadúca kvôli nedostatku informácií o systéme, o analyzovanej aktivite alebo kvôli ľudskému faktoru. Niekedy kvantitatívna analýza jednoducho nie je zaručená a vyžadovaná. V takomto prípade sú komparatívne polokvantitatívne a kvalitatívne ohodnotenia rizík, vykonané špecialistami vo svojom obore, stále veľmi efektívne. Aj v prípade uskutočnenia plnej kvantifikácie rizík však treba brať do úvahy fakt, že vypočítané úrovne rizík sú stále len odhadom. Kvantitatívna analýza sa obvykle vykonáva pri zložitejších projektoch, resp. pri komplexnejších rizikách a je úzko spätá s modelovaním a simuláciou vytvoreného modelu (kapitola 3.5.1) [13].

V kapitole 3.5 si stručne popíšeme niekoľko zaužívaných modelovacích techník, používaných v rámci analýzy rizík a predstavíme si tiež dve techniky, ktoré sú v tomto smere pomerne novinkou (Markovova analýza 3.5.7 a analýza Petriho sieťami 3.5.8).

3.4.3 Vyhodnotenie rizík

Etapa vyhodnotenia rizík je etapou porovnania odhadnutých úrovni rizík s kritériami, ktoré boli ustanovené v danom kontexte. Týmto spôsobom sa definuje celková úroveň rizika a jeho typ. Táto etapa je teda akýmsi premostením medzi porozumením analyzovaného rizika a rozhodnutiami o ďalšom smerovaní. Tieto rozhodnutia obvykle zahrňujú [13]:

- odpoveď na otázku, či je vôbec nutné dané riziko spracovať alebo akokoľvek naň reagovať
- priority spracovania jednotlivých rizík
- rozhodnutie, ktoré z možných scenárov nutno aplikovať

Posledným výstupom etapy vyhodnotenia rizík a vlastne celého rizikového posudku je dokumentácia, ktorá popisuje výsledky jednotlivých procesov tohto posudku. V nej obsiahnuté definované a analyzované riziká je samozrejme nutné zrozumiteľne popísať. Rozsah dokumentácie závisí na cieľoch a odbore posudku, avšak s výnimkou veľmi jednoduchých posudkov obvykle obsahuje [13]:

- ciele a pole pôsobnosti
- popis relevantných častí systému a jeho funkcií
- zhrnutie interného a externého kontextu organizácie a jeho súvis so situáciou, systémom alebo okolnosťami, ktoré sú posudzované
- aplikované kritériá rizík a ich zdôvodnenie
- limitácie, predpoklady a odôvodnenia hypotéz
- posudkovú metodológiu
- výsledky identifikácie rizík
- dáta, predpoklady, ich zdroje a spätnú validáciu
- výsledky analýzy rizík a ich vyhodnotenie
- analýzu citlivosti a neistoty
- kritické predpoklady a iné faktory, ktoré je nutné monitorovať
- diskusiu k dosiahnutým výsledkom
- referencie

3.5 Techniky a metódy analýzy rizík

Analýza rizík je jednou z najviac diverzifikovaných etáp celého manažmentu rizík. Dôvodom je práve výber a použitie rôznych techník a metód analýzy, ktorých je v dnešnej dobe už pomerne veľké množstvo. Výber techniky závisí samozrejme od zamerania projektu alebo aj od typu jednotlivých rizík. V praxi sa počas jednej analýzy rizík spravidla kombinuje hneď niekoľko metód a techník, či už kvalitatívnych, kvantitatívnych alebo obidvoch. V tejto kapitole si predstavíme niekoľko takýchto techník, pričom budeme klásť dôraz na Markovovu analýzu a analýzu Petriho sieťami.

3.5.1 Modelovanie a simulácia

Modelovanie a následná simulácia modelu nie je technikou analýzy rizík v pravom slova zmysle. Množstvo metód (spravidla kvantitatívnych) však zakladá na správnom vytvorení modelu na základe reality, ktorý je potom analyzovaný pomocou simulácie. Tieto dva pojmy sú teda v analýze rizík veľmi dôležité, preto im je venovaná celá táto sekcia.

Pre pochopenie konceptu modelovania a simulácií je nutné najprv osvojiť si nasledujúce pojmy [17]:

- **System**

System možno vo všeobecnosti definovať ako súbor elementárnych častí (tiež nazývanými *prvky systému*), medzi ktorými existujú určité väzby. Systémy možno deliť do niekoľkých kategórií podľa rôznych kritérií. Najčastejšie je delenie z hľadiska existencie na *reálne* a *fiktívne* a z pohľadu zmien stavov na *statické* a *dynamické*. V analýze rizík najčastejšie simulujeme práve reálne dynamické systémy.

- **Model**

Model je napodobenina systému iným systémom, ktorý je často zjednodušený a zobrazuje len časti objektívnej reality. Pre účely tvorcu modelu však musí napodobňovať všetky podstatné vlastnosti systému.

V prípade, že každý prvok modelu odpovedá jednému prvku jeho reálnej predlohy a naopak, hovoríme, že tento model je *izomorfný*. Vytvárať izomorfné modely je však veľmi obtiažne a prakticky to je možné len u veľmi jednoduchých predlohách reálneho systému. Ako príklad zložitosti vytvorenia izomorfného modelu možno uviesť napr. model jadrového reaktoru, kde je samozrejme veľmi dôležité, aby model (v praxi obvykle počítačový program) čo najvernejšie kopíroval reálnu predlohu. Z hľadiska enormnej komplexnosti tejto predlohy je však priam nemožné zostrojiť absolútne identický model systému. V praxi omnoho častejší je teda *homomorfný* model, kde každý prvok modelu odpovedá jednému prvku reálnej predlohy ale nie naopak. Vzniknutý model môže byť *fyzický*, čo je prípad, keď sa vytvorí akási hmotná kópia systému (napr. model vesmírnej stanice v hangári, kde budúci kozmonauti simulujú prácu a ovládanie stanice vo vesmíre), alebo *abstraktný*, resp. matematický. V druhom menovanom prípade je modelom napr. sústava rovníc [8].

- **Modelovanie**

Modelovanie je teda proces vytvárania modelu na základe našich znalostí o reálnej predlohe. Je to veľmi náročný proces, ktorý obvykle vyžaduje znalosti z viacerých oborov a kvalita tohto procesu (resp. kvalita výsledného modelu) zásadne ovplyvňuje výsledky, ktoré vzniknú experimentovaním s modelom.

- **Simulácia**

Tento termín už bol spomenutý v predošlých odsekoch a možno si pod ním predstaviť metódu získavania nových znalostí o reálnom systéme experimentovaním s jeho modelom. Ak chceme získať relevantné výsledky o reálnom systéme, je obvykle nutné opakovať simulačné experimenty viackrát a s rôznymi parametrami.

Pre naše potreby budeme teda simulovať výskyt rôznych typov rizík a ich celkový dopad na projekt, resp. organizáciu.

3.5.2 Riadená diskusia o riešení nových problémov (brainstorming)

Prvou metódou analýzy (v tomto prípade skôr len identifikácie) rizík, ktorú si stručne načrtneme je *brainstorming*. Ide o jednu z najstarších a veľmi rozšírených metód, ktorá nepotrebuje žiadne špeciálne nástroje alebo matematické aparáty, nakoľko sa jedná kvalitatívnu metódu.

Brainstorming možno v kontexte analýzy rizík chápať ako podnecovanie a povzbudzovanie k voľnej konverzácii v skupine kompetentných osôb, za účelom identifikácie rizík a atribútov s nimi spojenými. Termín „brainstorming“ je častokrát používaný veľmi voľne, znamenajúc akýkoľvek typ skupinovej diskusie. Avšak skutočný brainstorming zahŕňa presné techniky smerujúce k uisteniu sa, že predstavivosť diskutujúcich ľudí je v plnej pohotovosti. Nakoľko brainstorming kladie veľký dôraz na predstavivosť, je obzvlášť užitočný pri identifikácii rizík, spojených s novými technológiami (bez predošlých informácií a skúseností). Brainstorming môže byť formálny alebo neformálny. Formálny brainstorming je viac štrukturovaný a účastníci sú dopredu informovaní. Diskusia má tiež presne definovaný účel a idea výsledkov diskusie je predložená vopred [13].

Silnou stránkou tejto techniky je okrem spomínaného dôrazu na predstavivosť aj zainteresovanosť všetkých kompetentných osôb a pomerne veľká rýchlosť a jednoduchosť zriadenia. Problémy môžu nastať najmä v nedostatku odbornosti zúčastnených osôb a vo vzájomnom porozumení. Je zrejmé, že táto technika slúži v podstate len na identifikáciu rizík, v ich vyhodnotení už nemá prakticky žiadny význam [13].

3.5.3 Štrukturovaná „What if“ – technika

Pod uvedeným názvom sa skrýva systematická technika, založená na tímovej štúdiu. Jedným z hlavných atribútov tejto metódy je použitie priamych slov a fráz, ktoré podnecujú zainteresovaných k identifikácii rizík. Štandardné „what if“¹ – frázy sú teda kľúčovým elementom pri skúmaní, aký vplyv na systém, časť prevádzky, organizáciu alebo procedúry má odklon od zaužívaných operácií a správania. Pôvodne bola táto technika navrhnutá špeciálne pre štúdiu hazardov v chemických a petrochemických továrňach, dnes je však veľmi rozšírená v ostatných odvetviach. Vo všeobecnosti možno povedať, že skúma následky prípadných zmien a riziká takto zmenené, prípadné novovzniknuté. Skúmaná položka, systém alebo situácia je rozdelená na viacero kľúčových elementov, čo uľahčuje proces analýzy. Tieto elementy sú potom predmetom diskusie medzi pokiaľ možno všetkými zainteresovanými subjektami. Diskusia je vedená neustálym kladením „what if“ – otázok. Zámer je prediskutovanie všetkých scenárov, ktoré môžu nastať a tiež ich dopadov a príčin. Na posúdenie rizík sa tu obvykle používa kvalitatívna alebo polokvantitatívna metóda, pričom riziká sa hodnotia podľa priority. Podobne ako v prípade brainstormingu, aj táto metóda má značnú výhodu v rýchlosti a v minimálnych požiadavkách na prípravu. Aplikácie tejto metódy na vysokej úrovni však môže vykazovať malú detailnosť [13].

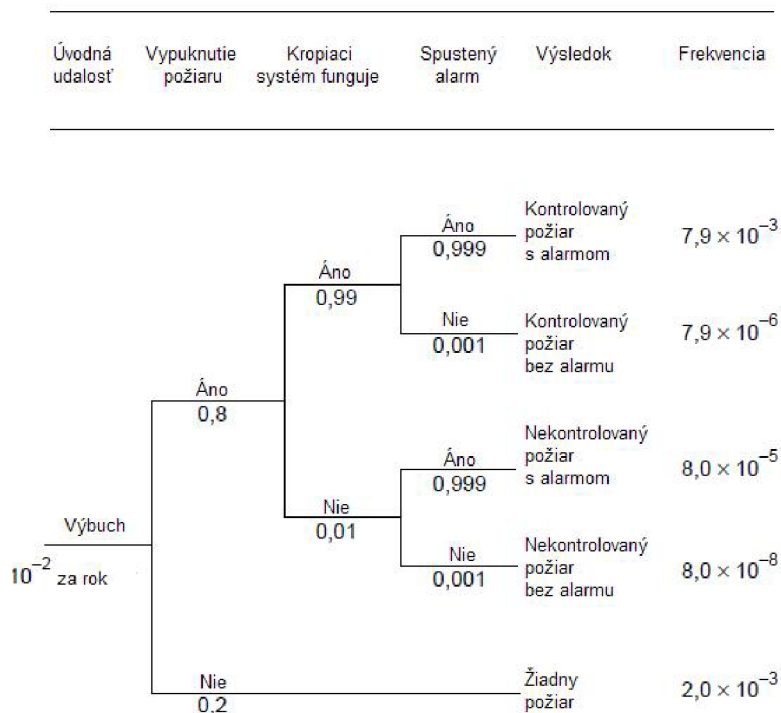
3.5.4 Analýza stromom udalostí

Analýza stromom udalostí² je grafická technika, používaná na reprezentáciu vzájomne vylučujúcich sa sledov rôznych udalostí. Strom udalostí začína koreňovou udalosťou, ktorá reprezentuje fungovanie, resp. nefungovanie nejakého systému. Analýza je navrhnutá na

¹ide o anglickú frázu, ktorá v preklade znamená „Čo ak“

²strom udalostí sa v praxi častejšie označuje anglickým ekvivalentom „event tree“

zmiernovanie dopadov rizík a môže byť aplikovaná kvalitatívne i kvantitatívne. Jednoduché kalkulácie v strome udalostí možno vidieť na obrázku 3.2 [13].



Obrázok 3.2: Príklad stromu udalostí (prevzaté z [13])

Analýzu stromom udalostí teda možno použiť na modelovanie, počítanie a ohodnocovanie (v kontexte rizík) rôznych nehodových scenárov, ktoré vyplývajú z nejakej počiatkovej udalosti. Metóda je použiteľná v akejkoľvek fáze životného cyklu produktu alebo procesu. Ak však chceme touto metódou dospieť ku kvalitnému rizikovému posudku, je nutné identifikovať všetky možné počiatkové udalosti. To je možné dosiahnuť pomocou iných metód (kapitola 3.5.2 a 3.5.3), avšak vždy môže nastať prípad prehliadnutia nejakej dôležitej počiatkovej udalosti [13].

3.5.5 Analýza rozhodovacím stromom

Rozhodovací strom³ reprezentuje rozhodovacie alternatívy a výstupy v sekvenčnom štýle. Týmto spôsobom sú vysvetlené výstupy a výsledky s istou úrovňou neistoty. Svojou štruktúrou je podobný stromu udalostí (kapitola 4). Rozhodovací strom totiž tiež začína počiatkovou udalosťou (alebo počiatkovým rozhodnutím) a modeluje rôzne cesty a výstupy. Tie sa odvíjajú od udalostí, ktoré môžu nastať alebo od rozhodnutí, ktoré môžu byť učené [13].

Analýza rozhodovacím stromom je v oblasti analýzy rizík veľmi významná technika. Okrem zobrazenia rizikových variant zohráva veľkú úlohu pri stanovení optimálnej rozhodovacej varianty v kontexte procesov s viacerými etapami. Graficky je rozhodovací strom realizovaný klasickým spôsobom ako postupnosť uzlov a hrán orientovaného grafu [18].

³rozhodovací strom sa v praxi častejšie označuje anglickým ekvivalentom „decision tree“

Veľkou výhodou rozhodovacieho stromu je bezpochyby jeho zrozumiteľná grafická reprezentácia a kalkulácia očakávaných hodnôt pre každú znázornenú cestu. V praxi sa však často objavujú negatívne tendencie zjednodušovania stromu, aby ho bolo možné znázorniť ako stromový diagram. V takom prípade ale už samozrejme scenár rizika nezodpovedá realite v takej miere, v akej by bolo vhodné [13].

3.5.6 Simulácia Monte Carlo

Mnoho systémov je príliš zložitých, aby sa úroveň ich neistoty a pravdepodobnosti vyjadrila analytickou metódou. Tieto systémy však môžu byť vyhodnotené na základe vstupu, ktorý pozostáva z náhodných premenných a počtu N výpočtov (simulácií), ktorých výsledkom je N možných výstupov požadovaného výsledku. Metóda Monte Carlo teda pokrýva ťažko pochopiteľné zložité situácie, na ktoré nemožno aplikovať analytické metódy. Keď bola táto technika uvedená prvýkrát, bol počet jej iterácií na tú dobu enormný a veľmi spomaľoval proces analýzy. V dnešnej dobe s omnoho vyspelejšou technológiou je však táto metóda jednou z najpoužívanejších [13].

Simulácia Monte Carlo teda zahŕňa náhodné ohodnotenie pravdepodobností rizík v rámci modelu, čím produkuje stovky až tisíce rôznych scenárov. Týmto získava táto metóda niekoľko značných výhod [21]:

- rozvrhnutie premenných modelu nemusí byť nijakým spôsobom odhadované
- môžu byť modelované korelácie a iné závislosti
- úroveň matematiky, potrebná na vykonanie simulácie Monte Carlo je pomerne jednoduchá
- všetky kalkulácie vykonáva počítač
- potrebný softvér je komerčne prístupný
- môže byť zahrnutá zložitejšia matematika, pričom sa úroveň zložitosti modelu veľmi nezvyší
- simulácia Monte Carlo je široko uznávaná ako presvedčivá technika
- skúmanie správania sa modelu je jednoduché a názorné
- zmeny môžu byť vykonané rýchlo, pričom výsledky možno okamžite porovnať s predošlým modelom

Hlavnou nevýhodou tejto metódy je závislosť presnosti výsledku od počtu simulácií (v dnešnej dobe rýchlych počítačov je však táto nevýhoda zanedbateľná). Obrovské a zložité modely tiež môžu byť veľkou výzvou pre analytika a zainteresované osoby môžu mať problém spojiť si model s reálnym procesom [13].

3.5.7 Markovova analýza

Markovova⁴ analýza spolu s analýzou pomocou Petriho sietí (kapitola 3.5.8) je kľúčovou analýzou tejto diplomovej práce, v rámci ktorej tiež bola vyvinutá aplikácia, podporujúca túto metódu.

⁴pomenovaná po Andrejovi Markovovi (1856 – 1922), ruskom matematikovi

Markovova analýza sa používa v prípade, keď budúci stav systému závisí len na jeho súčasnom stave. Metóda môže byť rozšírená aj na zložité systémy použitím Markovových procesov vyššieho rádu a je striktno vymedzená na model, matematické výpočty a odhady. Proces Markovovej analýzy je kvantitatívna technika, ktorá môže byť diskretná (použitím pravdepodobností zmien medzi stavmi) alebo spojitá (použitím intenzity zmien naprieč stavmi). Metóda môže byť kalkulovaná ručne, avšak ako oveľa výhodnejšie sa javí použitie počítačových programov, ktorých už na trhu existuje niekoľko. Technika Markovovej analýzy môže byť použitá na viacero systémových štruktúr, zahrňujúc [13]:

- nezávislé paralelné komponenty
- nezávislé sériové komponenty
- systémy so zdieľanou záťažou
- systémy v pohotovosti (zahrňujúc prípady, keď môže nastať porucha prepínania)
- degradované systémy

Vstupom pre Markovovu analýzu je zoznam rôznych stavov, do ktorých sa systém, subsystém alebo komponenta môžu dostať (napr. plne funkčný stav, degradovaný stav, chybový stav). Ďalej je nutné absolútne porozumenie možných prechodov medzi stavmi, ktoré sú pri modelovaní nevyhnutné [13].

Markovove techniky sú obzvlášť užitočné pri skúmaní procesov so zálohovaním alebo systémov, pri ktorých porucha systému závisí na postupnosti udalostí. Ďalej je túto metódu možno úspešne aplikovať na systémy so zložitými stratégiami údržby (systémy s prioritami obnovy, s problémami radenia do fronty a s obmedzenými zdrojmi). Model samozrejme musí primerane odrážať reálny systém s ohľadom na dané stratégie a politiky údržby. V takomto modeli je potom možné uvažovať aj poradie, v ktorom sa vyskytujú viacnásobné poruchy systému. Markovove techniky sú z teoretického hľadiska pomerne pružné a všestranne použiteľné. Ich aplikácia do praxe však musí prebiehať obzvlášť obozretne, nakoľko počet stavov a prechodov rýchlo vzrastá s počtom prvkov systému. To môže (a v praxi často vedie) k chybám modelu a nesprávnym zázorneniam. [12].

Kľúčovým pojmom Markovovej analýzy a to hlavne v tejto diplomovej práci je **Markovov reťazec**. Ide o pravdepodobnostný model, v ktorom nasledujúci model pokusu závisí výlučne na výsledku predchádzajúceho pokusu (ako už bolo načrtnuté na začiatku tejto kapitoly). Túto vlastnosť nazývame *Markovova vlastnosť* (anglicky *Markov property*) alebo tiež „bezpamätovosť“. Hlavným konceptom tohto modelu je teda určenie stavov systému a prechodov medzi nimi, pričom platí, že sa jedná o náhodnú postupnosť [8].

Markovove reťazce pracujú s diskretnými stavmi v diskretnom čase⁵. Možno ich tiež rozdeliť na *homogénne* a *nehomogénne*. Homogénne reťazce sú charakterizované konštantnými hodnotami prechodov medzi stavmi, pričom v prípade nehomogénnych reťazcov sa hodnoty pravdepodobností prechodov menia. [9].

V tejto práci sa budeme zaoberať výlučne homogénnymi Markovovými reťazcami.

V rámci Markovových reťazcov je popísaných niekoľko typov stavov s rôznymi vlastnosťami. Najdôležitejšie z nich sú nasledovné [8]:

- *trvalý stav*: systém sa do tohto stavu vracia s pravdepodobnosťou 1

⁵v prípade spojitých stavov by išlo o *Markovov proces*

- *prechodný stav*: pravdepodobnosť návratu do tohto stavu je menšia než 1
- *absorpčný stav*: prechody v rámci systému v konečnom dôsledku vedú do tohto stavu (z tohto stavu už teda nie je možné dostať sa inam)
- *trvalý nulový stav*: počet krokov pre návrat do tohto stavu je rovný nekonečnu
- *periodický stav*: návrat do tohto stavu je možný len po určitej perióde m
- *ergodický stav*: je trvalý, nenulový a neperiodický

Obvyklý zápis matice prechodov homogénneho Markovovho reťazca je nasledovný:

$$\mathbf{P} = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{pmatrix}$$

Dôležitým pojmom je tiež *vektor absolútných pravdepodobností* stavov v okamihu m :

$p(m) = (p_1(m), p_2(m), \dots, p_n(m))$, $m = 0, 1, 2, \dots$ obsahuje pravdepodobnosti, že systém je v danom okamihu v stave S_1, S_2, \dots, S_n .

Vektor počiatočných pravdepodobností popisuje stav systému v okamihu jeho vzniku ($m = 0$). Zadaním tohto vektoru a stochastickej matice prechodov je Markovov reťazec plne definovaný, nakoľko platí vzťah [8]:

$$p(m) = (p(m-1)) \bullet P = p(0) \bullet P^m, m = 0, 1, 2, \dots$$

V mnohých prípadoch sa absolútne pravdepodobnosti po dosiahnutí určitého počtu krokov prestanú významne líšiť, teda dosiahnu akúsi limitnú hodnotu. Hovoríme, že Markovov reťazec sa ustálil. Tieto limitné pravdepodobnosti možno vypočítať podľa vzťahu [8]:

$$p^\infty = B^{-1}b$$

kde p^∞ je vektor limitných pravdepodobností, B je báza matice a b je vektor pravých strán.

Výpočet je odvodený od sústavy rovníc, ktoré počítajú pravdepodobnosti jednotlivých stavov o n neznámych stavoch (pre maticu s n riadkami a n stĺpcami). Neznáme stavy sú zostavené do matice, ktorá po vynásobení jednotkovou maticou E normalizuje pomyselnú pravú stranu na samé nuly. Ľavé strany rovníc tvoria jednotlivé riadky novovzniknutej matice. Pri výpočte sa jedna z týchto rovníc vynechá (kvôli lineárnej závislosti) a nahradí rovnicou [8]:

$$1 = p_1 + p_2 + \cdots + p_n$$

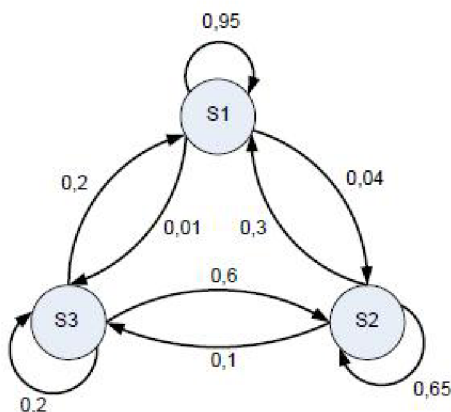
Takýmto spôsobom možno nájsť spomínaný vektor limitných pravdepodobností, pričom konkrétny príklad takéhoto výpočtu si ukážeme v ďalšom texte.

Markovova analýza je teda zameraná na koncept stavov (napr.: „k dispozícií“, „porucha“, atď.) a prechodov medzi nimi, ktoré sú založené na konštantnej pravdepodobnosti zmeny. Na popis prechodov sa používa stochastická prechodová matica pravdepodobností, ktorá umožňuje kalkuláciu viacerých výstupov. Na elementárnu ilustráciu techniky Markovovej analýzy budeme uvažovať systém, ktorý môže nadobúdať jeden z troch stavov: funkčný, degradovaný, poruchový (definované ako S_1 , S_2 , S_3). Každý deň existuje systém práve v jednom z týchto stavov. Tabuľka 3.5 znázorňuje pravdepodobnosť zajtrajšieho stavu systému [13].

		Zajtrajší stav		
		S1	S2	S3
Dnešný stav	S1	0,95	0,04	0,01
	S2	0,3	0,65	0,05
	S3	0,2	0,6	0,2

Tabuľka 3.5: Markovova matica (prevzaté z [13])

Znázornené pole pravdepodobností sa nazýva Markovova matica, prípadne matica prechodov. Všimnime si, že súčet hodnôt každého riadku sa rovná 1, čo reprezentuje súčet všetkých možných výstupov v každom prípade. Systém tiež možno reprezentovať Markovovým diagramom 3.3, kde stavy systému znázorňujú kruhy a prechody zasa šípky s príslušnou hodnotou pravdepodobnosti [13].



Obrázok 3.3: Príklad Markovovho diagramu (prevzaté z [13])

Nech P_i reprezentuje pravdepodobnosť, s akou sa systém môže ocitnúť v stave i pre $i = 1, 2, 3$. Potom možno zapísať sústavu rovníc [13]:

$$P_1 = 0,95P_1 + 0,30P_2 + 0,20P_3$$

$$P_2 = 0,04P_1 + 0,65P_2 + 0,60P_3$$

$$P_3 = 0,01P_1 + 0,05P_2 + 0,20P_3$$

Uvedené rovnice nie sú nezávislé a požadované tri neznáme nevypočítajú. Musí byť použitá nasledujúca rovnica a jedna z predošlých sa musí zahodiť [13].

$$1 = P_1 + P_2 + P_3$$

Výsledky sú 0,85, 0,13 a 0,02 analogicky pre stavy 1, 2, 3. Systém je teda plne funkčný 85% svojho času, v degradovanom stave je 13% času a v stave poruchy je 2% celkového času [13].

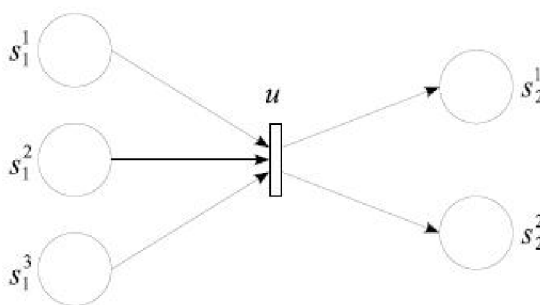
3.5.8 Analýza Petriho sieťami

Pod pojmom Petriho⁶ sieť si možno predstaviť širokú triedu diskretných matematických modelov, ktoré umožňujú popisovať riadiace toky a informačné závislosti v rámci modelovaných systémov. V súčasnej dobe sú Petriho siete najčastejšie spájané s modelovaním paralelných a distribuovaných systémov a to rovnako v oblasti hardvéru i softvéru. Najtypickejšie využitie tohto matematického aparátu je teda v oblasti paralelných architektúr, komunikačných protokolov, paralelných databázových systémov, prekladačov a počítačových sietí. Petriho siete sú samozrejme veľmi perspektívne aj v iných odvetviach, napr. v telekomunikáciách, pri popise automatizovaných priemyselných systémov alebo v administratívnych systémoch [23].

Petriho sieť môže byť definovaná viacerými spôsobmi, pričom jedným z nich je matematicko – sémantický model. Ten sa však v praxi príliš nepoužíva a ustupuje do úzadia pred praktickejším a názornejším grafickým modelom, ktorý umožňuje vizualizáciu dynamiky systému. Tento spôsob je najvhodnejší pre modelovanie rôznych systémov a analýz, tým pádom samozrejme aj pre analýzu rizík [15].

Petriho siete vznikli rozšírením klasických konečných automatov o ďalšie modelovacie možnosti. Pri použití konečného automatu je základným popisovacím prostriedkom *stav* a *prechod* medzi stavmi. Teraz predpokladajme, že je účelné popísať nejaký stav automatu (napr. S_1) určitými podmienkami či parciálnymi stavmi (napr. S_1^1, S_1^2, S_1^3). Analogicky pre iný stav: napr. S_2 popíšeme parciálnymi stavmi S_2^1, S_2^2 [23].

Túto situáciu je potom možné prostriedkami Petriho sietí vyjadriť ako je uvedené na obrázku 3.4.



Obrázok 3.4: Modelovanie zmeny stavu Petriho sieťou (prevzaté z [23])

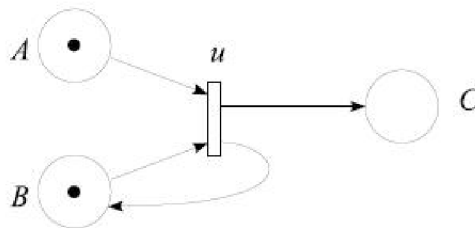
Znázornená distribúcia stavu systému na parciálne stavy, ktoré sú obvykle vyjadriteľné nejakými podmienkami a spojenie týchto podmienok s udalosťami systému je základným jadrom popisu diskretného systému Petriho sieťou. Parciálne stavy sa graficky zobrazujú kružnicou a odborne ich nazývame *miesta Petriho siete*. Udalosti sú zasa zobrazované

⁶pomenovaná po Carlovi Adamovi Petrim (1926 – 2010), nemeckom matematikovi

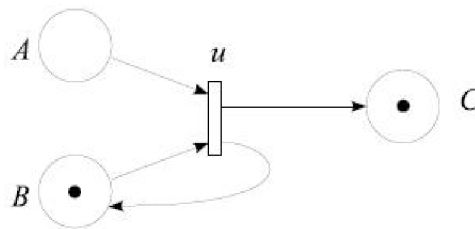
ako obdĺžniky, prípadne úsečky a nazývajú sa *prechody Petriho siete*. Systém na obrázku 3.4 teda možno interpretovať tak, že udalosť u môže nastať v prípade, keď sú splnené podmienky, odpovedajúce miestam S_1^1, S_1^2, S_1^3 . Táto udalosť následne spôsobí, že systém dosiahne stavu, pri ktorom platia podmienky, vyjadrené miestami S_2^1, S_2^2 [23].

Okamžitý stav modelovaného systému je teda v Petriho sieti vyjadrený určitými parciálnymi stavmi, s príslušnými miestami siete. Na definovanie dosiahnutia istého parciálneho stavu, ktorý odpovedá príslušnému miestu siete slúži tzv. *značenie* miesta. Jedná sa o nezápornú celočíselnú informáciu, ktorá je graficky reprezentovaná určitým počtom bodiek – *značiek*, ktoré sú umiestnené vo vnútri príslušného miesta. Aby mohla prebehnúť určitá udalosť modelovaná prechodom, musí existovať v každom vstupnom mieste prechodu spomínaná značka [23].

Toto chovanie si ilustrujeme na obrázkoch 3.5 a 3.6.



Obrázok 3.5: Udalosť môže nastať (prevzaté z [23])



Obrázok 3.6: Udalosť prebehla (prevzaté z [23])

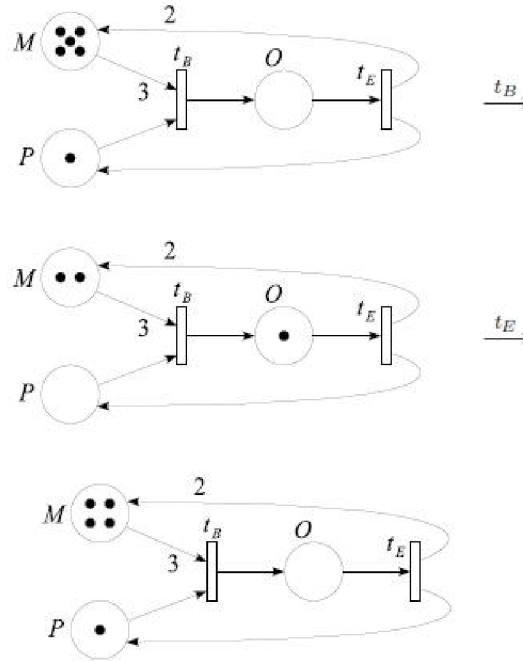
V predošlom odseku sme si značenie definovali ako celočíselnú nezápornú informáciu. Z toho vyplýva, že vstupnú podmienku pre nejakú udalosť nemusíme nutne definovať (ne)existenciou jedinej značky, ale vo všeobecnejšej verzii Petriho sietí aj vopred definovaným minimálnym počtom značiek. Týmto spôsobom sa možno veľmi dobre modelovať obmedzené zdroje systému [23].

Ako ilustráciu takého prípadu uvažujme istú operáciu počítača, ktorej vykonanie vyžaduje voľný procesor (miesto P) a pridelenie troch pamäťových blokov (miesto M). Po ukončení operácie sú dva pamäťové bloky uvoľnené a vrátené k ďalšiemu použitiu. Máme teda definované:

- M : počet voľných pamäťových blokov
- P : procesor (nie) je voľný
- O : operácia (ne)prebieha

- t_B : začiatok operácie
- t_E : koniec operácie

Celočíselné ohodnotenie hrany určuje počet značiek, ktorý sa odoberá, resp. pridáva do určitých miest. S touto konvenciou si ilustrujeme definovaný modelom na nasledujúcom obrázku 3.7.



Obrázok 3.7: Vykonalenie prechodov Petriho siete (prevzaté z [23])

Ďalší dôležitý fakt, týkajúci sa značenia miesta siete je jeho kapacita. Tá môže byť neobmedzená alebo obmedzená. Obmedzenú kapacitu vyjadrujeme celým kladným číslom, ktoré udáva maximálny počet značiek v danom mieste. Kapacitné obmedzenie sa využíva predovšetkým v situáciách, keď chceme vylúčiť preplnenie (napr. pamäte alebo fronty) [23].

Po predošlých informáciách o Petriho sieťach už môžeme na tomto mieste pristúpiť k ich najvšeobecnejšej definícii [23]:

Definícia 3.5.1. Trojicu $N = (P, T, F)$ nazývame *sieťou*, ak:

- P je konečná množina elementov, ktoré sa nazývajú *miesta* siete N
- T je konečná množina elementov, ktoré sa nazývajú *prechody* siete N , pričom $P \cap T = \emptyset$
- F je binárna relácia, nazývaná *toková relácia* siete N , pre ktorú platí $F \subseteq (P \times T) \cup (T \times P)$

Grafom siete nazývame biparitný orientovaný graf, ktorý vznikne grafovou reprezentáciou relácie F . Množina $P \cup T$ je množinou vrcholov grafu siete. Miesta sú obvykle kreslené v tvare kružníc a prechody v tvare obdĺžnikov alebo úsečiek.

Definícia 3.5.1 je elementárnym matematickým popisom Petriho sietí, ktorých sa však postupom času vyvinulo niekoľko druhov. Nové typy Petriho sietí sú definované zložitejšie, resp. rozširujú základnú definíciu o ďalšie modelovacie prvky. Z rôznych variácií Petriho sietí možno spomenúť napr. najjednoduchšie *C/E siete*, o niečo zložitejšie *P/T siete* a konečne i *vysokourovňové Petriho siete* (sem patria napr. aj farebné *Petriho siete*). Práve hlavne vysokourovňové Petriho siete zažívajú v poslednom období veľký rozmach vo veľkom množstve rôznych odvetví. Ich koncept, definície a grafické notácie popisuje norma *ISO/IEC15909 – 1* [15].

Medzinárodný štandard Petriho sietí vysokej úrovne je aplikovateľný na širokú škálu systémov so súbežnými diskretnými udalosťami. V kontexte analýzy rizík sa môže jednať o bezpečnostnú analýzu, identifikáciu scenárov s vysokým rizikom a modelovanie toku dát. Konkrétnejšie je v rámci organizácie možné uvažovať napr. o systémoch automatickej kontroly, manufaktúrach a komunikačných sieťach. V uvedených prípadoch je analýza pomocou Petriho sietí extrémne vhodná, pretože je názorne vizualizovaná synchronizácia, zdieľanie zdrojov a správanie sa súbežných procesov [15].

Táto kapitola je len absolútnym základom pre elementárne pochopenie princípu Petriho sietí. Jedná sa totiž o veľmi komplexnú problematiku, ktorá niekoľkonásobne presahuje rozsah tejto publikácie a ktorá vyžaduje veľmi odborné a špecializované znalosti.

3.6 Markovova analýza verzus analýza Petriho sieťami

V predošlých kapitolách 3.5.7 a 3.5.8 sme si stručne definovali základnú problematiku Markovových techník a Petriho sietí. V oboch prípadoch (zvlášť v prípade Petriho sietí) sa jedná o komplexné matematické aparáty, ktoré sa používajú na modelovanie a analýzu systémov, ktoré vykazujú chovanie závislé na stavoch týchto systémov. Z toho vyplýva, že najlepšie uplatnenie týchto techník je najmä v oblasti zálohovacích mechanizmov, poruchovosti systémov a stratégie údržby. Petriho siete navyše veľmi dobre vizualizujú paralelizmus procesov, čo ich predurčuje k úspechu v oblasti modelovania sieťovej komunikácie, manufaktúry, automatickej kontroly a v iných oblastiach, kde je nutné znázornenie zložitých interakcií medzi prvkami systému.

Norma *IEC61165* [12] poukazuje na fakt, že zvláštna trieda Petriho sietí – všeobecné stochastické Petriho siete majú ekvivalentnú spôsobilosť modelovania ako Markovove techniky. Petriho siete možno považovať za prirodzený implicitný výraz ich reprezentácie pomocou explicitného Markovovho modelu. Záver tohto zistenia je, že Petriho siete možno previesť na Markovove modely. Modely všeobecných stochastických Petriho sietí síce často obsahujú pomerne zložité interakcie, ale diagram, ktorým sú popísané je väčšinou omnoho menší, kompaktnější a názornejší ako je tomu v prípade Markovových techník. V praxi sa pre účely vyhodnotenia prevádza Petriho sieť na odpovedajúci Markovov model, ktorý sa potom analyzuje. To sa obvykle deje pomocou softvérových nástrojov.

3.6.1 Aplikácia modelovacích techník v manažmente rizík IT projektov

Markovova analýza i analýza Petriho sieťami sú pomerne nekonvenčné metódy analýzy rizík v manažmente IT projektov a v manažmente projektov všeobecne. Je dôležité podotknúť že uvedené dve i všetky ostatné metódy analýzy rizík sa takmer nikdy nepoužívajú výlučne. Hlavne u rozsahovo väčších projektov je to v podstate vylúčené. Akýkoľvek projekt je v celku pomerne komplexný systém, ktorý z podstaty vyžaduje hybridné modely, pretože jedinou technikou možno popísať celý systém len pri veľkej úrovni abstrakcie. To však obvykle

nekorešponduje s reálnymi výsledkami a analýza rizík môže v takom prípade postrádať zmysel. Zvlášť uvedené dve analýzy sa používajú na modelovanie menších podsystémov, v ktorých je možné dobre definovať stavy a prechody. Výsledné analytické zistenia potom slúžia ako vstupné hodnoty iným technikám, ktoré berú jednotlivé riziká a systém, resp. projekt ako taký z globálnejšieho hľadiska.

Hlavne v prípade Markovových techník sa často uvádza, že môžu tvoriť sériu podsystémov v rámci *analýzy stromu poruchových stavov*⁷, ktorý má podobnú štruktúru ako rozhodovací strom alebo strom udalostí (kapitoly 4 a 3.5.5). V takomto prípade sa za niektorými hradlami stromu v skutočnosti skrývajú Markovove modely, ktoré sa vyhodnocujú samostatne. Výsledný model sa často nazýva hybridná, prípadne dynamická analýza FTA [12].

Manažment rizík IT projektov je potenciálne veľmi perspektívna oblasť pre obidve porovnané analýzy. Oproti iným projektom majú IT projekty totiž okrem ekonomických, legislatívnych, riadiacich a mnohých iných rizík aj potenciálne veľké množstvo rizík technického rázu. Práve v tejto vetve rizík nájdeme spomínané analýzy najväčšie uplatnenie.

3.6.2 Zvolenie modelovacej techniky na implementáciu aplikácie

V predošlých odsekoch sme spomenuli nespornú výhodu Petriho sietí v grafickej reprezentácii modelovaného systému, ktorá je obvykle omnoho názornejšia ako je tomu v prípade Markovových modelov. Čo sa však týka zložitosti jednotlivých problematík, Petriho siete vyžadujú omnoho väčšie množstvo skúseností a vedomostí. Pre mnohých manažérov rizík v tomto prípade nastáva problém, pretože aj po intenzívnom naštudovaní danej problematiky môže nastať problém pri vytvorení obzvlášť komplexného modelu zložitého systému. Rozdiel zložitostí Markovových techník a Petriho sietí v konečnom dôsledku vidno už na samotných matematických definíciách jednotlivých modelovacích aparátov. To platí dvojnásobne u vysokoúrovňových Petriho sieťach.

Nakoľko sme v predošlej kapitole dospeli k záveru, Petriho siete možno previesť na Markovov model a teda majú ekvivalentnú výpovednú hodnotu, bude sa aplikácia, ktorá je súčasťou tejto diplomovej práce zaoberať práve Markovovou analýzou. Hlavným cieľom vyvíjanej aplikácie bude, aby manažéri rizík všetkých odvetví (teda nie len IT) boli po krátkom naštudovaní problematiky schopní vytvárať užitočné modely, ktoré budú prínosom kvantitatívnej analýzy rizík ich projektov.

⁷v praxi sa často používa názov FTA, odvodený od anglického „Fault Tree Analysis“

Kapitola 4

Analýza a návrh aplikácie

Cieľom tejto kapitoly je popis analýzy a návrhu systému, ktorý bude implementovaný ako podporná aplikácia kvantitatívnej analýzy rizík v manažmente projektov. V predošlej kapitole sme pre tento účel uprednostnili Markovovu analýzu pred analýzou pomocou Petriho sietí. Na nasledujúcich riadkoch bude popísaná neformálna špecifikácia systému, návrh grafického rozhrania a reprezentácie uložených dát.

Počas analýzy tejto problematiky som sa zoznámil s komerčnými aplikáciami ModelRisk od Vose Software [5] a FaultTree++ od firmy Isograph [2].

ModelRisk je považovaný za jeden z najprofesionálnejších nástrojov analýzy rizík. Jedná sa v podstate o rozšírenie produktu Microsoft Excel, preto je nutné mať nainštalovaný balík Microsoft Office. ModelRisk teda pracuje priamo v tabuľkovom prostredí programu Excel a ponúka širokú škálu rôznych techník a metód analýzy rizík, s hlavným zameraním na vysokorýchlostnú metódu Monte Carlo (kapitola 3.5.6). Výsledky analýzy možno pohodlne štatisticky vyhodnotiť a názorne vizualizovať. ModelRisk tiež obsahuje Markovovu analýzu, konkrétne modelovanie Markovových reťazcov, pre ktorú je vstupom tabuľka pravdepodobností, resp. matica prechodov.

FaultTree++ je program zameraný na 3 typy analýz: analýza stromu poruchových stavov (FTA), analýza stromom udalostí (ETA), ktorá je v tejto práci stručne predstavená v kapitole a konečne Markovova analýza. Táto aplikácia zvolila iný prístup ako ModelRisk a to priamu grafickú vizualizáciu modelu. Jedná sa o desktopovú aplikáciu, kde užívateľ priamo v grafickom prostredí skladá požadovaný model, pričom jednotlivým stavom a prechodom definuje požadované atribúty.

V tejto kapitole predstavíme návrh aplikácie, ktorá kombinuje niektoré výhody oboch spomínaných komerčných produktov a dáva im nový rozmer. Výsledná aplikácia je zrozumiteľná a názorná a užívateľ bude schopný už po krátkej štúdií problematiky zostavovať Markovove modely intuitívnou formou.

4.1 Neformálna špecifikácia

Už v kapitole o Markovovej analýze sme naznačili, že aplikácia bude pracovať s homogénnymi Markovovými reťazcami. Markovov reťazec pracuje v diskretnom čase, čo je v rovine manažmentu projektov absolútne dostačujúce, nakoľko v praxi kontrola stavov systému, resp. projektu tiež prebieha v diskretných okamihoch (napr. týždenné overenie stavu projektu oproti plánu atď.).

Prvým dôležitým rozhodnutím smerovania aplikácie je spôsob zadávania dát Mar-

kovovho modelu, kde máme na výber medzi dvomi možnosťami. Prvou je vytvorenie stochastickej matice prechodov, druhou je interaktívne vytváranie Markovovho diagramu v grafickom prostredí. V prospech diagramu hovorí hlavne jeho názornosť, čo je bezpochyby veľké pozitívum. Faktom však je, že s rastúcim počtom stavov a prechodov sa výrazne degraduje prehľadnosť diagramu a už v prípade desiatich stavoch sa v ňom užívateľ môže nadobro stratíť. Z tohto dôvodu bude teda užívateľ definovať práve prechodovú maticu, pričom aplikácia bude podporovať aj automatické generovanie príslušného diagramu.

Výsledky analýzy by mali byť zobrazené v prehľadnej podobe v ďalšej matici, ktorá bude reprezentovať pravdepodobnosti prechodov po určitom počte krokov analýzy. Takisto je veľmi žiadúci tiež vizuálny priebeh zmeny pravdepodobností pomocou čiarového grafu. Jednotlivé stavy bude možné pomenovať a v krátkosti tiež popísať. V Markovovej analýze sa často vyskytuje aj takzvaný *vstupný vektor*, ktorý sa naplňa nezápornými celočíselnými hodnotami. Tieto hodnoty sa potom v rámci analýzy asociujú s pravdepodobnosťami prechodov medzi jednotlivými stavmi. Funkciu tohto vektoru si popíšeme na ilustratívnom príklade v nasledujúcich kapitolách.

4.2 Spôsob uloženia dát

Ďalším dôležitým rozhodnutím v kontexte smerovania aplikácie je spôsob uloženia dát. V prípade, že by aplikácia slúžila zároveň ako komplexné úložisko registrov rizík a vyžadovala by nutnosť správy viacerých typov užívateľov, najvhodnejšou voľbou by bolo využitie relačnej databázy a jazyka SQL. V našom prípade však aplikácia slúži exkluzívne na analytické vyhodnotenie Markovových modelov a žiadna diverzifikácia typov užívateľov nie je potrebná. Rovnako i spomínané komerčné nástroje nevyužívajú databázové servery a všetky potrebné dáta ukladajú do súborov na lokálny počítač. Vynechanie databázových služieb tiež uľahčuje prácu manažéra rizík (ktorý je primárnym užívateľom aplikácie) v zmysle, že nemusí podstupovať rôzne procedúry nastavení, ktoré sú spojené s používaním relačných databáz.

Formát súboru, ktorý bude využívať naša aplikácia je XML. XML je skratka z anglického „**eXtensible Markup Language**“ a znamená *rozšírený značkovací jazyk*. Je to na systéme nezávislý jazyk pre reprezentáciu dát a ich štruktúry v *dokumente XML*. Dokument XML je textový súbor znakov Unicode, ktorý uvedené dáta obsahuje spolu so značkami, ktoré definujú ich štruktúru. Nakoľko sa jedná o textový dokument, je možné ho vytvoriť v akomkoľvek (nie nutne špecializovanom) textovom editore. Presnú definíciu XML má na svedomí konzorcium World Wide Web Consortium (W3C) a možno ju nájsť na webovej adrese [ODKAZ](#). Dokument XML je primárne určený na prenos dát z jedného počítača do druhého. Dokumenty XML sú teda spravidla tvorené a spracovávané počítačovými programami. Na rozdiel od jazyka HTML, kde je sada značiek (tagov) pevne daná, XML je jazyk, ktorý umožňuje definovať vlastné sady tagov a atribútov tak, aby vyhovovali rôznym druhom dát. XML je teda často definovaný ako *metajazyk*, inými slovami ako jazyk na definovanie iných jazykov [11].

4.2.1 Štruktúra dokumentu v jazyku XML

Dokument v XML pozostáva z dvoch hlavných častí, z **prológu (hlavičky)** a **tela dokumentu**. Ich voľná definícia je nasledovná [11]:

- Prológ poskytuje informácie, ktoré sú nevyhnutné pre interpretáciu obsahu v tele

dokumentu. Obsahuje dve voliteľné časti a keďže sú obe voliteľné aj samotný prológ je tým pádom voliteľný. Uvedené dve komponenty prológu, ktoré sa musia nachádzať v poradí, v akom sú uvedené sú:

- Deklarácia XML, ktorá definuje verziu XML a môže tiež určovať konkrétnu sadu v Unicode a fakt, či je dokument samostatný alebo nie.
 - Deklarácia typu dokumentu určuje externú definíciu typu dokumentu (DTD). Tá obsahuje deklaráciu značenia elementov v tele dokumentu, alebo priamo obsahuje explicitnú deklaráciu značenia, alebo pozostáva z obidvoch možností.
- Telo dokumentu obsahuje samotné dáta. Pozostáva z jedného alebo niekoľkých elementov. Každý element je pritom definovaný počiatočným a koncovým tagom. Tieto elementy definujú štruktúru dát, pričom vždy existuje jediný *koreňový element*, ktorý zastrešuje všetky ostatné elementy.

4.2.2 Návrh XML jazyka pre vyvíjanú aplikáciu

Nasledujúca schéma 4.1 pomerne intuitívne ozrejmuje XML jazyk, ktorý bol navrhnutý pre potreby implementovanej aplikácie.

```
<MarkovMatrix>
  <ID1 name="ok" description="the device is fully operational" inputvector="20">
    <ID1 prob="0.94"/>
    <ID2 prob="0.05"/>
    <ID3 prob="0.01"/>
  </ID1>
  <ID2 name="failure" description="the device is currently out of order" inputvector="2">
    <ID1 prob="0.3"/>
    <ID2 prob="0.65"/>
    <ID3 prob="0.05"/>
  </ID2>
  <ID3 name="dead" description="the device is incontrovertibly broken" inputvector="0">
    <ID1 prob="0.0"/>
    <ID2 prob="0.0"/>
    <ID3 prob="1.0"/>
  </ID3>
</MarkovMatrix>
```

Obrázok 4.1: Ilustrácia navrhnutého XML jazyka na fiktívnom Markovovom modeli

Univerzálnym koreňovým elementom navrhnutého jazyka je `<MarkovMatrix>`. Za ním nasleduje element, ktorý definuje súčasný stav, ktorého atribúty sú: meno, popis a príslušná hodnota vstupného vektoru. Tú možno na uvedenom príklade interpretovať tak, že práve disponujeme dvadsiatimi zariadeniami, ktoré sú plne funkčné a dvomi dočasne pokazenými. Podelementami súčasného stavu sú stavy budúce, ktorých atribút `prob` značí pravdepodobnosť prechodu zo súčasného do budúceho stavu.

4.3 Prípady použitia

Už v neformálnej špecifikácii aplikácie bolo naznačené, že užívatelia aplikácie nebudú nijak špeciálne diverzifikovaní. V praxi bude používateľom pravdepodobne manažér rizík projektu, avšak pre naše potreby definovania prípadov použitia môžeme predpokladať všeobecného, bližšie nešpecifikovaného užívateľa. Prípady použitia, ktoré sa vzťahujú na implementovanú aplikáciu sú teda nasledovné:

- zadanie nového Markovovho modelu (tomu predchádza zadanie počtu stavov modelu)
- načítanie existujúceho modelu
- uloženie rozpracovaného modelu
- vykonanie Markovovej analýzy v zadanom okamihu diskrétného času
- zobrazenie grafu priebehu Markovovej analýzy v zadanom časovom rozmedzí
- zobrazenie automaticky vygenerovaného Markovovho diagramu
- zobrazenie pomocníka
- zmena atribútov jednotlivých stavov
- zmena rozsahovej škály pravdepodobností
- ukončenie aplikácie

4.4 Návrh grafického rozhrania

Vyvíjaný program bude mať podobu desktopovej aplikácie. Hlavné okno bude obsahovať prechodovú maticu s implicitne nulovými pravdepodobnosťami prechodov a rovnako vynulovaný vstupný vektor. V spodnej časti okna sa budú nachádzať informácie o súčasnom a budúcom stave, ktoré sa budú dynamicky meniť, v závislosti od toho, na akú bunku tabuľky (matice prechodov) užívateľ klikne. Dynamicky sa tiež bude meniť súčet pravdepodobností jednotlivých riadkov, ktorý bude závisieť od užívateľského vstupu. Hlavné okno bude obsahovať klasický panel a päť tlačidiel, z ktorých každé otvorí nové okno. Ich funkcia bude nasledovná:

- úprava informácií o jednotlivých stavoch
- aplikovanie hodnôt vstupného vektoru na pravdepodobnosti prechodov
- zobrazenie matice prechodov po zadanom počte krokov Markovovej analýzy
- zobrazenie grafu stavov systému v zadanom časovom úseku
- vykreslenie Markovovho diagramu

Kapitola 5

Implementácia

V tejto kapitole sa budeme zaoberať samotnou transformáciou návrhu z predošlej kapitoly na funkčnú aplikáciu. Na začiatku si definujeme implementačný jazyk a vývojové prostredie. V ďalšej fáze si popíšeme niekoľko algoritmov, ktoré bolo nutné v rámci implementácie navrhnuť, pričom budeme klásť dôraz na oddelenie implementačných častí grafického rozhrania od riadiacich častí a dátových častí. Dôkladne popíšeme funkcie aplikácie, pričom tento popis bude podporený snímkami samotnej aplikácie. V závere demonštrujeme funkčnosť a prínos aplikácie na niekoľkých Markovových modeloch.

5.1 Implementačné nástroje

Na vytvorenie aplikácie bol použitý programovací jazyk Java v prostredí NetBeans IDE 6.8, ktorý je veľmi vhodný na tvorbu desktopových aplikácií. Využitie boli najmä metódy balíka Java JDK 6 ale tiež voľne dostupné a slobodne šíriteľné balíky JCommon a JFreeChart [3] (metódy na tvorbu grafov) a tiež balík dom4j [1] (manipulácia s XML dokumentami). Nakoľko sú všetky uvedené nástroje voľne šíriteľné, vzniká možnosť prípadného ďalšieho funkčného rozšírenia inými užívateľmi.

5.1.1 Jazyk JAVA

Java je moderný multiplatformový jazyk, čo značí, že programy v ňom vytvorené sú prenositeľné medzi známymi operačnými systémami. Java predovšetkým umožňuje písať malé programy, ktoré nazývame **applety**. Jedná sa o programy, ktoré možno začleniť do webových stránok a dodať im tak istú inteligenciu. Táto možnosť prináša širokú radu možností.

Java ale tiež umožňuje vývoj rozsiahlych aplikačných programov, čo je vlastnosť, ktorá je využitá aj v tejto diplomovej práci. Prenositeľnosť jazyka Java je jednou z najdôležitejších vlastností tohto jazyka od počiatku jeho vývoja. Aplikácia v Jave teda vyžaduje len jedinú sadu zdrojového kódu, bez ohľadu na radu rôznych počítačových platforiem. Takmer v každom inom programovacom jazyku sa často vyžadujú úpravy zdrojového kódu, aby sa vyhovelo rôznym počítačovým prostrediam (obzvlášť v prípade náročných grafických užívateľských rozhraní).

Ďalšia dôležitá vlastnosť jazyka Java je, že sa jedná o jazyk **objektovo orientovaný**. Objektovo orientované programy sú obvykle zrozumiteľnejšie a menej náročné na čas pri ich údržbe a rozširovaní. Java tiež implicitne podporuje rôzne znakové sady.

Java je jazyk, ktorý sa učí pomerne rýchlo, ale faktom je, že ho je pomerne veľa. To znamená, že aby bol programátor schopný v Jave efektívne programovať, bude potrebovať porozumieť veľmi obsiahlym knižniciam, ktoré k tomuto jazyku patria. Na ich zvládnutie však existuje celá rada návodov a príkladov na webových stránkach a tiež množstvo knižných publikácií [11].

5.1.2 NetBeans IDE 6.8

Prostredie NetBeans je veľmi komplexné, prehľadné a hlavne voľne dostupné programovacie prostredie, ktoré podporuje jazyk Java. Užívateľ je schopný s prehľadom kontrolovať štruktúru zdrojového kódu, pričom je mu za istých okolností dynamicky ponúkaný výber metód jednotlivých tried spolu s priloženou dokumentáciou (tzv. JavaDoc), čím sa zefektívňuje proces programovania a odpadá nutnosť neustále vyhľadávať názvy a funkcionality požadovaných metód a tried v externých zdrojoch.

Prostredie NetBeans tiež disponuje prehľadným debuggerom a editorom na tvorbu grafických rozhraní. V súčasnosti už existuje verzia 7.0, ktorá je voľne stiahnuteľná z oficiálnych webových stránok tohto projektu [4].

5.1.3 Knižnica JfreeChart

Základné knižnice jazyka Java neobsahujú nástroje, pomocou ktorých by sa efektívne a pohodlne dali vytvárať grafy. Samozrejme to je v konečnom dôsledku možné implementovať si vlastné triedy, ktoré budú založené na implicitných grafických primitívach, avšak taká implementácia by bola náročná a hlavne zbytočná.

Vo februári roku 2000 bol totiž Davidom Gilbertom založený projekt JFreeChart, ktorý je presne určený pre potreby zobrazenia grafov. Jedná sa o voľne šíriteľnú knižnicu, ktorá je kompletne napísaná v jazyku Java. Ponúka širokú škálu typov grafov a ich zobrazenie je skutočne na profesionálnej úrovni. Knižnica je distribuovaná pod licenciou GNU LGPL a je možné ju voľne stiahnuť z oficiálnej webovej stránky [3].

V implementácii bola použitá aktuálne najnovšia verzia knižnice – JFreeChart 1.0.13

5.1.4 Knižnica dom4j

Na základe požiadaviek uvedených v návrhu systému bolo v implementácii nutné vytvárať a spracovávať dokumenty XML. Veľmi efektívny nástroj na tieto účely je ďalšia voľne šíriteľná knižnica – dom4j. Tá okrem spracovania XML dokumentov ponúka aj radu ďalších funkcií, ktoré nás však v rámci tohto projektu nebudú zaujímať.

V implementácii bola použitá verzia dom4j 1.6.1, ktorú možno voľne stiahnuť z oficiálnej webovej stránky [1].

5.2 Hlavné funkcie aplikácie

V nasledujúcich podkapitolách si podrobne popíšeme všetky hlavné funkcionality systému. Popisy tiež budú podporené radou snímok, ktoré budú znázorňovať okno aplikácie v určitých situáciách. Uvidíme, že implementácia plne korešponduje s návrhom aplikácie (kapitola 4) vo všetkých bodoch (grafické rozhranie, reprezentácia dát, atď.). Vďaka prenositeľnosti jazyka Java je možné aplikáciu spustiť pod operačným systémom Microsoft Windows i pod systémom UNIX, prípadne MAC OS. Aplikácia nevyžaduje pripojenie k

internetu ani pripojenie na databázový server. Celá aplikácia „komunikuje“ s užívateľom v anglickom jazyku.

5.2.1 Hlavné okno a kalkulácia Markovovho reťazca

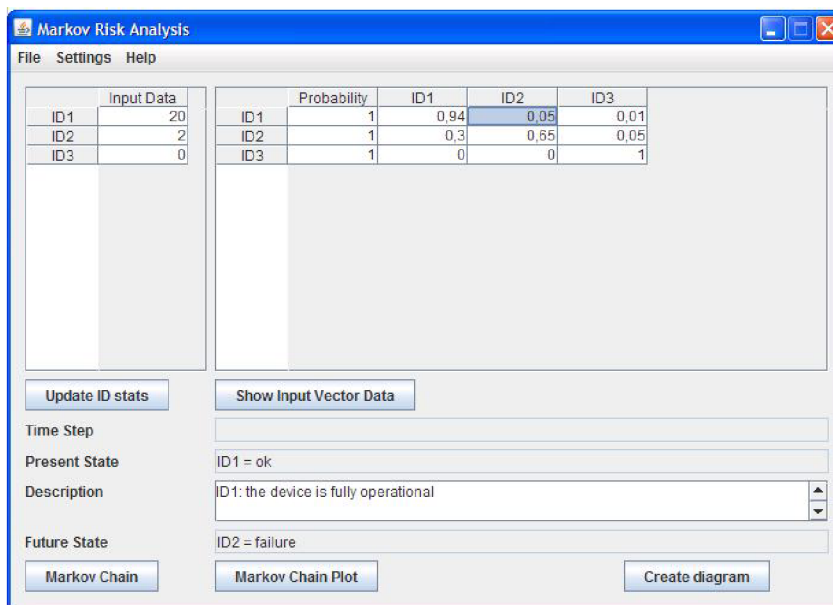
Po spustení aplikácie sa nám v strede obrazovky zobrazí okno s lištou a prázdny obsahom. Lišta tohto hlavného okna obsahuje 3 elementy: **File**, **Settings** a **Help**.

Položka **File** sa v podstate ničím zásadným nelíši od podobných položiek v iných desktopových aplikáciách. Obsahuje možnosť začatia nového Markovovho modelu a ukončenia aplikácie. Okrem toho samozrejme umožňuje uložiť rozpracovaný model, resp. takýto model kedykoľvek načítať.

Pod možnosťou **Settings** sa skrýva voľba tzv. „škály pravdepodobností“. Užívateľ si takto môže zvoliť, či chce zadávať pravdepodobnosti do matice prechodov v intervale 0 až 1 alebo 0 až 100. V závislosti od toho potom jednotlivé riadky matice prechodov musia tvoriť súčet 1, resp. 100. Položka Settings poskytuje miesto pre ďalšie nastavenia v prípadnom budúcom vývoji aplikácie (napr. dynamické pridávanie a uberanie stavov a pod.).

Položka **Help** opäť zodpovedá konvenciám iných desktopových aplikácií. Možno sa tu dozvedieť základné informácie o programe (názov, autor, rok výroby) a tiež informácie o funkcionalite programu.

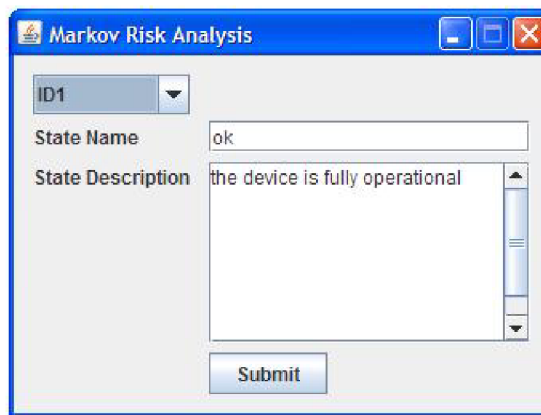
V prípade zvolenia nového modelu v sekcii File nás program vyzve, aby sme zadali počet stavov modelu. Nakoľko sa aplikácia venuje len analýze rizík (kapitola 3.4.2) a nie ich identifikácií (kapitola 3.4.1), musí byť počet stavov modelovaného systému analytikovi už v tejto fáze známy. Ak by sme sa rozhodli modelovať príklad, ktorý sme si uviedli v kapitole 4.2.2, na ktorom bola ilustrovaná štruktúra navrhnutého XML jazyka, zadali by sme počet požadovaných stavov – 3. Po vyplnení všetkých atribútov, ktoré XML dokument uvádza by hlavné okno aplikácie vyzeralo tak ako vidíme na obrázku 5.1.



Obrázok 5.1: Hlavné okno aplikácie

Ako môžeme vidieť, aplikácia naozaj zodpovedá stanoveným požiadavkam a návrhu. V efektívnom priestorovom rozložení máme vedľa seba maticu prechodov so zadaným počtom stavov i vstupný vektor. Obe spomenuté entity sú užívateľom modifikovateľné intuitívnym spôsobom.

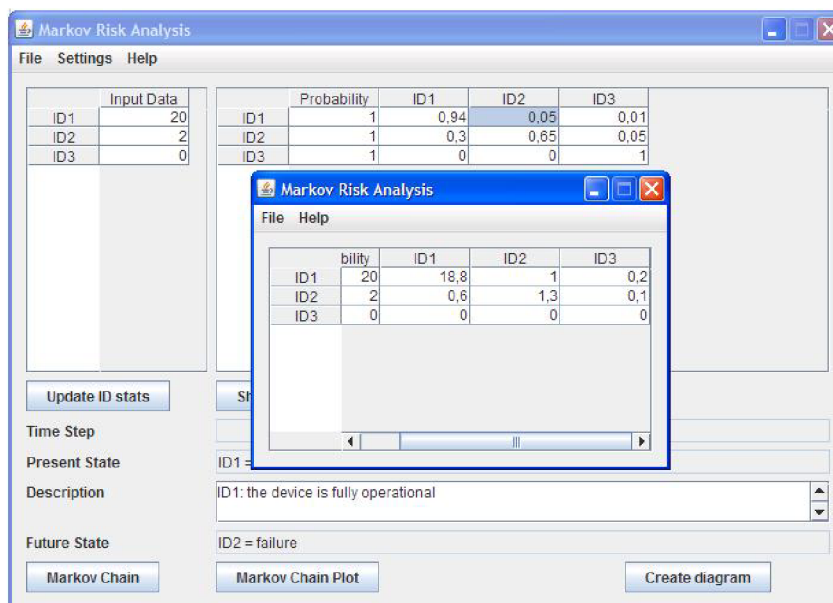
V spodnej časti poľa sa nachádzajú 3 polia, ktoré užívateľ priamo z hlavného okna modifikovať nemôže. Ide o pomenovanie súčasného stavu (riadky matice) a jeho popis a tiež pomenovanie budúceho stavu (stĺpce matice). Implicitne sú tieto hodnoty po vytvorení nového modelu nastavené na **no name** pre pomenovanie stavu a **no description** pre popis stavu. Na obrázku 5.1 však vidíme, že tieto hodnoty už užívateľ modifikoval. Zobrazenie týchto hodnôt sa dynamicky mení v závislosti od toho, ktorú bunku tabuľky užívateľ označí. V príklade na obrázku je označená pravdepodobnosť prechodu zo stavu 1 do stavu 2, resp. z „ok“ do „failure“. Na modifikáciu týchto textových polí slúži tlačítko **Update ID stats**, ktoré vytvorí v popredí hlavného okna nové okno, znázornené na obrázku 5.2.



Obrázok 5.2: Úprava údajov jednotlivých stavov

V tomto okne už atribúty jednotlivých stavov modifikovať môžeme. V ľavom hornom rohu okna si zvolíme požadovaný stav a v textových poliach vyplníme želané údaje. Pre uloženie každého jednotlivého stavu je nutné zakaždým stlačiť tlačidlo **Submit** (ak užívateľ najprv vypní všetky údaje a až potom stlačí požadované tlačidlo, tak sa uloží len aktuálne zobrazený stav a ostatné ostatné bez zmeny).

V hlavnom okne sme si všimli, že vo vstupnom vektore už sú zadané nejaké hodnoty, pričom by bolo vhodné vidieť rozdelenie týchto hodnôt na základe zadaných pravdepodobností v matici prechodov. Tento účel spĺňa tlačidlo **Show Input Vector Data**. V prípade, že by toto tlačítko bolo aktivované v okamihu, aký je ilustrovaný na obrázku hlavného okna (5.2), teda s hodnotami vo vstupnom vektore 20 2 a 0, vyzeral by výstup ako na obrázku 5.3.



Obrázok 5.3: Rozdelenie hodnôt zo vstupného vektora do matice prechodov

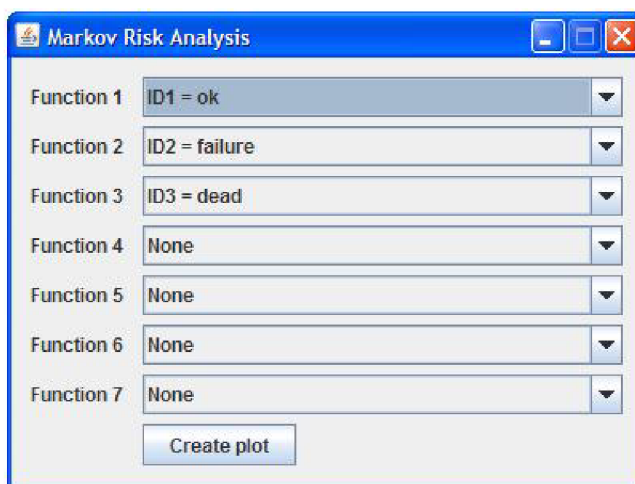
Vidíme teda, že sa znovu vytvorilo ďalšie okno, ktoré kopíruje maticu prechodov, tentokrát ale s analogicky dosadenými hodnotami vstupného vektora. Takto môžeme vidieť rozdelenie zdrojov (hodnôt vektora) v budúcom stave Markovovho modelu.

V tomto momente sa konečne dostávame k samotnému Markovovmu reťazcu. Po stlačení tlačidla **Markov Chain** v ľavom dolnom rohu hlavného okna sa nám zobrazí výzva pre zadanie počtu krokov Markovovho reťazca. To sa však udeje len za predpokladu, že všetky riadky prechodovej matice majú súčet 1, resp. 100. V opačnom prípade vyskočí okno s chybovou hláškou. V prípade, že s korektnou maticou prechodov zadáme počet krokov rovný 1, bude výsledkom identická matica prechodov, teda stav systému v budúcnosti po jednom kroku (čo vlastne reprezentuje samotná matica prechodov). Až po zadaní väčšieho počtu krokov sa dočkáme nového rozdelenia pravdepodobností v rámci stavov systému, ktoré už možno analyticky skúmať. Nakoľko je výsledkom tejto operácie matica, nemožno hovoriť celkom doslovne o Markovovom reťazci (kde je výsledkom vektor). Zobrazená matica však môže mať pre analytika väčšiu výpovednú hodnotu a okrem toho je z nej možné Markovov reťazec vydedukovať pomyselným dosadením jednoduchého počiatočného vektora. Pre uvedený príklad napr. platí, že ak by sme dosadili počiatočný vektor $[1, 0, 0]$, tak výsledný Markovov reťazec by bol vlastne prvý riadok vypočítanej matice. Zadaný vektor značí, že v súčasnosti sa modelovaný systém nachádza v stave 1, resp. v stave „ok“. Takýmto počiatočným vektorom budeme tiež implicitne predpokladať pri tvorbe grafu. Vypočítaná matica sa samozrejme zobrazí v novom okne, ktoré je takmer identické s hlavným oknom, takže nie je potreba ho znovu ilustrovať. Hlavný rozdiel oproti hlavnému oknu spočíva v absencii všetkých tlačidiel (okrem tlačidla *Show Input Vector Data*).

5.2.2 Vykreslenie grafu

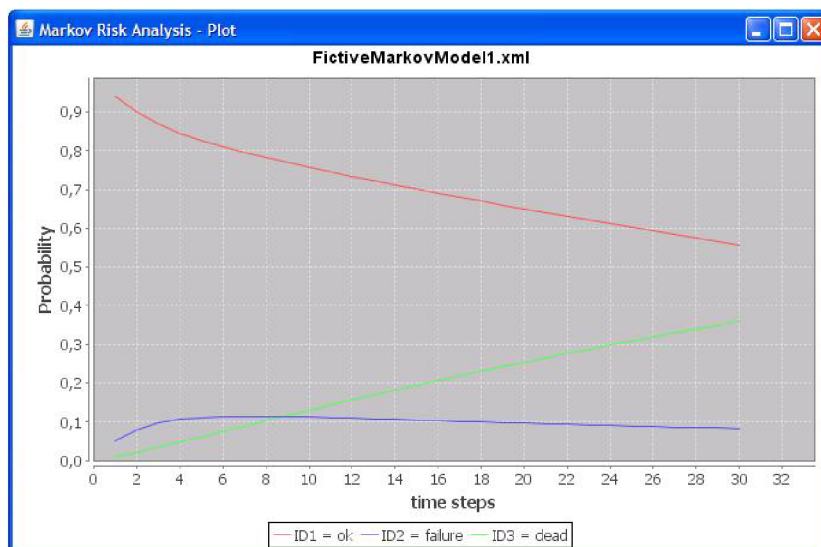
V tejto kapitole sa už budeme zaoberať samotným vykreslením grafu Markovovho reťazca v zadanom rozmedzí krokov. Už v predošlom odseku sme naznačili, že počiatkový vektor tu bude vždy definovaný rovnako (vieme totiž, že Markovov reťazec je plne popísaný počiatkovým vektorom a maticou prechodov), a to tak, že sa systém bude v súčasnosti nachádzať v prvom stave. To znamená, že napr. pre model s piatimi stavmi bude mať počiatkový vektor $[1, 0, 0, 0, 0]$.

Graf vykresľujeme tlačidlom **Markov Chain**. Ako v prípade tlačidla *Markov Chain* i tu najprv prebieha kontrola súčtu pravdepodobností riadkov matice prechodov. V korektnom prípade aplikácia vyzve užívateľa aby zadal požadovaný rozsah, resp. počet krokov Markovovho reťazca. Následne sa otvorí okno, zobrazené na obrázku 5.4.



Obrázok 5.4: Nastavenie grafu

Na uvedenom obrázku je opäť zobrazený náš ilustratívny príklad s tromi stavmi. Vidíme, že najvyšší počet funkcií, ktoré možno zobraziť je 7. Vyšší počet funkcií v grafe je už značne neprehľadný. V našom prípade však máme k dispozícii len 3 funkcie, nakoľko má model len 3 stavy. V tomto okne si môžeme zvoliť ich poradie, prípadne sa môžeme rozhodnúť, že vizualizáciu zmien pravdepodobností niektorého zo stavov vynecháme. Za predpokladu, že ponecháme odporúčané rozloženie z obrázku, vygenerujeme tlačidlom **Create plot** nasledujúci graf 5.5:



Obrázok 5.5: Graf vývinu Markovovho reťazca

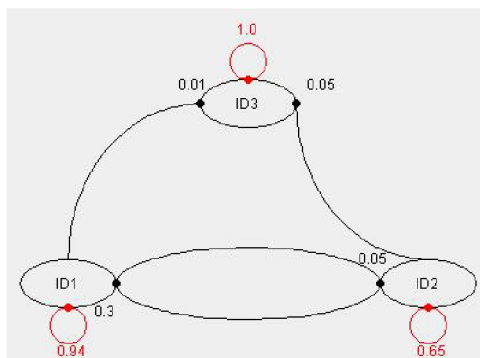
V horeuvedenom prípade sme si zvolili Markovov reťazec s počtom krokov 30. Počiatočný vektor (hodnoty z bodu 0 do bodu 1 na vodorovnej osi) nie je pre lepšiu prehľadnosť grafu znázornený.

Markovov reťazec je diskretný systém, preto by po správnosti mal byť generovaný graf bodový a nie spojitý. Takto má však užívateľ o niečo lepší prehľad, ako sa stav systému vyvíja.

5.2.3 Generovanie Markovovho diagramu

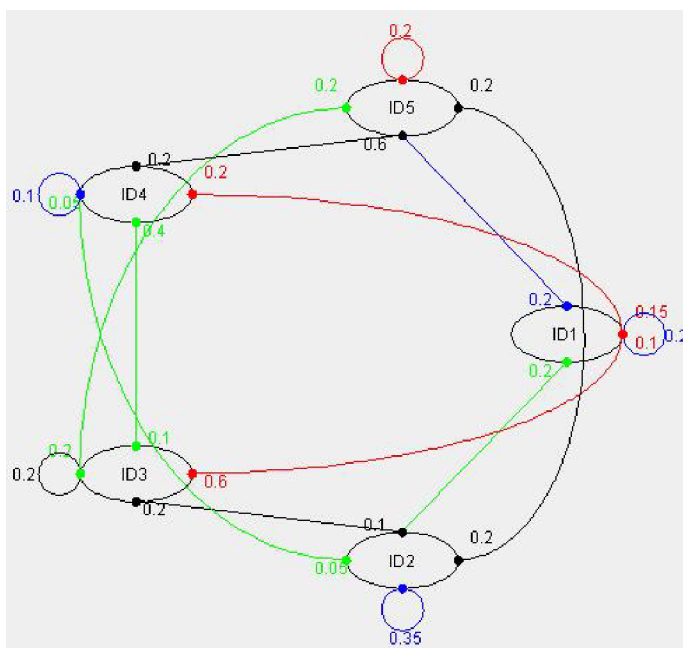
V kapitole analýzy a návrhu aplikácie sme si kládli otázku, či je prehľadnejšia a názornejšia matica prechodov alebo Markovov diagram. Dospeli sme k záveru, že v prípade menších systémov (model s menej stavmi) je prehľadnejší práve diagram. Avšak čím je systém zložitejší, jeho diagram neúmerne rýchlo stráca prehľadnosť. Implementovaná aplikácia umožňuje toto porovnanie kedykoľvek vykonať.

Po vytvorení alebo načítaní modelu môžeme pomocou tlačidla **Create diagram** kedykoľvek v priebehu modelovania vidieť príslušný vygenerovaný diagram. Pre náš hotový známy model sa v novom okne vygeneruje diagram znázornený na obrázku 5.6.



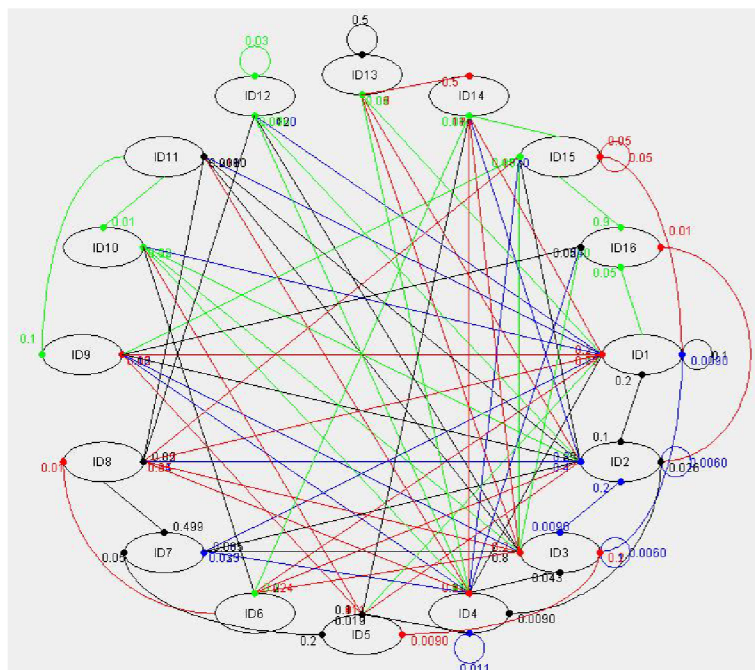
Obrázok 5.6: Diagram modelu o troch stavoch

Generovanie diagramu prebieha na základe viacerých implementovaných algoritmov, ktoré sa snažia o čo najväčšiu optimalizáciu výsledného diagramu a jeho prehľadnosti. Prechody medzi stavmi v systéme o počte stavov 4 a menej sú znázornené na krivkách zvlášť. V prípade modelu s väčším počtom stavov je už prechod zo stavu 1 do stavu 5 a naspäť (teda zo stavu 5 do 1) znázornený na jednej krivke. Tento prípad ilustruje napr. nasledujúci fiktívny model o piatich stavoch 5.7:



Obrázok 5.7: Diagram modelu o piatich stavoch

Nasledujúci model (obrázok 5.8) už ilustruje model o šestnástich stavoch a užívateľ si sám môže urobiť obraz, či je v takomto prípade prehľadnejšia matica prechodov, alebo diagram.



Obrázok 5.8: Diagram modelu o šestnástich stavoch

5.3 Ilustrácia funkčnosti aplikácie na vybraných Markovových modeloch

Už viackrát v priebehu celej práce boli vymenované hlavné odvetvia, kde sa Markovova analýza využíva najčastejšie. Spomenuli sme si, že táto analýza je v rámci manažmentu projektov vhodná obvykle na analýzu menších podsystémov a jej výstupy sú potom vstupmi pre iné analytické metódy, pre projekt ako celok.

Takýmto podsystémom môže byť aj náš fiktívny ilustratívny trojstavový model, ktorý nás sprevádza takmer celou druhou polovicou tejto diplomovej práce a ilustrovali sme si na ňom funkčnosť implementovanej aplikácie. Pre manažéra rizík má najväčšiu výpovednú hodnotu práve graf vývoja Markovovho reťazca, v tomto prípade obrázok 5.4. Výsledky grafu a jeho analytické vyhodnotenie môžeme interpretovať nasledujúcim spôsobom. Predstavme si, že tento model predstavuje nejaké zariadenie, ktoré je dôležité pre chod procesov v rámci projektu. Na začiatku máme k dispozícii 20 zariadení, ktorú sú v poriadku a 2, ktoré majú poruchu (hodnoty vstupného vektora). Ďalej si predstavme, že jeden „krok“ v našom modeli predstavuje časový úsek 1 týždeň. Ukončenie projektu sa plánuje o 30 týždňov a za celú túto dobu potrebujeme vždy aspoň 5 týchto zariadení, aby fungovali súčasne. Z uvedeného grafu vieme vyčítať, že v tridsiatom kroku (tridsiatom týždni) je predpokladaný stav systému taký, že viac ako 50% týchto zariadení bude fungovať (za tento čas sa niekoľko z nich nenávratne pokazí), čo je v prepočte viac ako 10 týchto zariadení. Nie je teda nutné v priebehu projektu dokupovať ďalšie zariadenia a predpokladané riziko nehrozí.

Na nasledujúcich modeloch sa pokúsime aplikovať Markovovu analýzu nie len na určité podsystémy a podprocesy v rámci nejakého projektu ale na projekt ako celok. Celý projekt si totiž možno predstaviť ako systém, ktorý nadobúda v diskretnom čase nejaké stavy. V

diskrétnom preto, lebo aj v praxi sa kontroly stavu prebiehajúceho projektu vykonávajú obvykle skokovo (napr. sumarizovanie výsledkov na konci každého týždňa). Markovov model kalkuluje predpokladaný stav systému v budúcnosti, čo môže byť pri analýze projektu mimoriadne užitočné.

Autor Jan Doležal vo svojej dizertačnej práci „Predikce v projektech s využitím Markovských řetězců“ [8] predstavil všeobecný model projektu s pomerne veľkou úrovňou abstrakcie, ale zároveň s dostatočne veľkou úrovňou zložitosti, aby potenciálne mohol byť využitý v praxi. Model má 8 stavov, ktoré sú definované nasledovne:

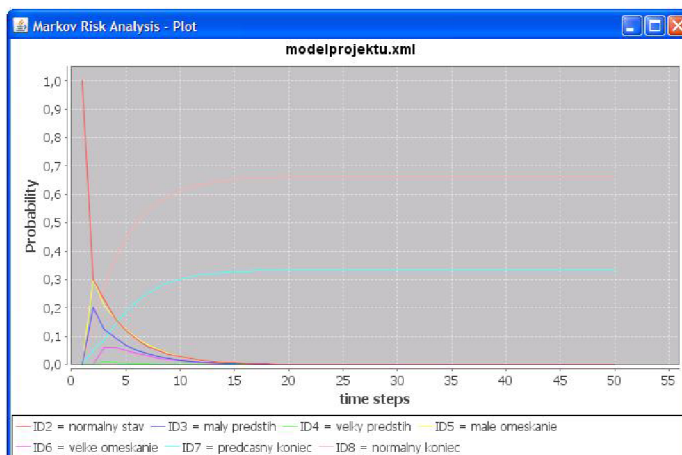
- **S** – štart – stav, z ktorého sa vychádza a je v podstate nadbytočný (použije sa len raz).
- **NS** – normálny stav – projekt beží podľa plánu
- **MP** – malý predstih – projekt je v miernom predstihu
- **VP** – veľký predstih – projekt je vo veľkom predstihu
- **MO** – malé oneskorenie – projekt má malé oneskorenie
- **VO** – veľké oneskorenie – projekt je veľmi oneskorený
- **PK** – predčasný koniec – projekt je z nejakých dôvodov ukončený skôr ako dosiahol vytýčené ciele
- **NK** – normálny koniec – projekt naplnil svoje ciele (nezisťujeme však, či zároveň skončil včas a v medziach rozpočtu)

Model bude obsahovať dva absorpčné stavy: predčasný a normálny koniec. Do týchto stavov je možné dostať sa z akýchkoľvek iných stavov (s výnimkou stavu **S**). Zámerne nebudeme uvažovať prechody medzi stavmi „veľké oneskorenie“ a „normálny stav“, resp. medzi „veľké predstihom“ a „normálnym stavom“. Ďalej sa predpokladá, že zotrvanie v predstihu je menej pravdepodobné ako zotrvanie v oneskorení. Pravdepodobnosti prechodov sú teda na základe týchto predpokladov definované nasledovne podľa tabuľky 5.1.

		<i>Budúci stav</i>							
		S	NS	MP	VP	MO	VO	PK	NK
<i>Súčasný stav</i>	S	0	1	0	0	0	0	0	0
	NS	0	0,3	0,2	0	0,3	0	0,05	0,15
	MP	0	0,4	0,25	0,05	0,15	0	0,05	0,1
	VP	0	0	0,7	0,1	0	0	0,05	0,15
	MO	0	0,2	0,05	0	0,3	0,2	0,05	0,2
	VO	0	0	0	0	0,2	0,3	0,4	0,1
	PK	0	0	0	0	0	0	1	0
	PK	0	0	0	0	0	0	0	1

Tabuľka 5.1: Matica prechodov pre model všeobecného projektu (prevzaté z [8])

Teraz sa opäť pomocou implementovanej aplikácie pokúsime vytvoriť graf podľa vzťahu pre absolútne pravdepodobnosti zotrvania v jednotlivých stavoch systému (počiatočný vektor je opäť implicitne $[1, 0, 0, 0, 0, 0, 0, 0]$). Priebeh štartovacieho stavu vynecháme a zameriame sa na zvyšných 7. Výsledok vidíme na obrázku 5.9.



Obrázok 5.9: Grafická reprezentácia výpočtu pre 50 krokov

Výsledok možno interpretovať tak, že približne po dvadsiatich krokoch je 33% pravdepodobnosť, že projekt skončí predčasne a 67% že skončí riadne. Podľa všetkého sa zároveň jedná o limitné pravdepodobnosti zotrvania v stavoch tohto Markovovho reťazca. Ak si teda predstavíme projekt s dĺžkou trvania cca. 20 týždňov (jeden krok je v tomto prípade 1 týždeň), tak výsledok analýzy zodpovedá všeobecným štatistikám o úspešnosti projektu. To pôvodne ani nebol zámer tohto modelu, ale názorne vidíme, že aj s prostým úvodným odhadom pravdepodobností prechodov medzi stavmi možno dosiahnuť relevantných výsledkov.

Po ilustrácii modelu všeobecného projektu, môžeme prejsť ku konkrétnejšiemu modelu, pričom sa zameriame špeciálne na IT projekt. Predošlý model pojednával o riziku nedokončenia cieľov a má len akúsi štatistickú funkciu. Preto z hľadiska analýzy rizík má len malý význam. V rámci projektu však existuje množstvo iných typov rizík, spojených napr. s časom, nákladmi, atď. V nasledujúcom texte si definujeme vlastný model IT projektu, na ktorý aplikujeme Markovovu analýzu v snahe zistiť, či je daný projekt možné dokončiť včas a s akou pravdepodobnosťou. Budeme teda skúmať časový aspekt.

Predpokladajme prípad, že sme v počiatočnej fáze IT projektu a na termín jeho dokončenie je o 270 dní. Pravidelnú kontrolu stavu projektu si určíme na každý tretí deň (to bude zároveň krok v Markovovej analýze). Stav projektu si definujeme nasledovne, pričom v tomto modeli nebudeme uvažovať, že projekt skončí v nejakej fáze predčasne:

- **IF** – inicializačná fáza – vytvorenie tímu, určenie manažéra a pod.
- **KZ** – komunikácia so zákazníkom – zistenie požiadaviek zákazníka
- **AP** – analýza požiadaviek
- **SP** – vypracovanie špecifikácie požiadaviek – na základe získaných poznatkov sa vypracuje neformálna špecifikácia požiadaviek

- **P** – tvorenie plánu – organizačná etapa, vytvorenie rozvrhu práce
- **N** – návrh aplikácie – formálny návrh aplikácie
- **I** – implementácia – implementácia aplikácie podľa návrhu
- **T** – testovanie – testovanie a ladenie vytvorenej aplikácie
- **O** – optimalizácia – optimalizácia aplikácia na základe výsledkov testov
- **PZ** – predanie produktu zákazníčkovi
- **U** – ukončenie projektu

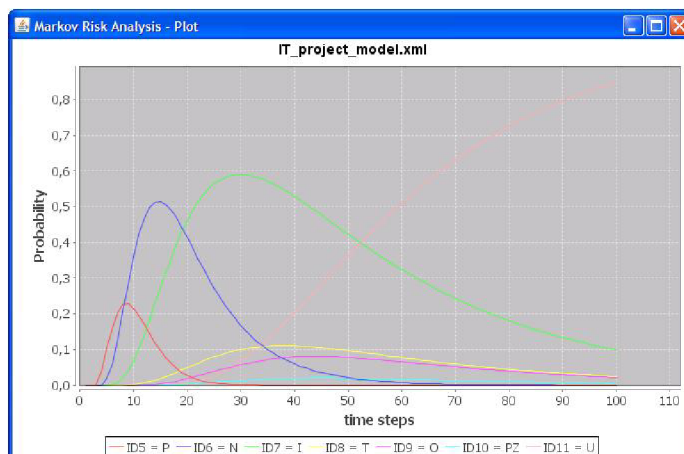
Náš vytvorený model IT projektu má teda 11 stavov. Manažérovi rizík v praxi nič nebráni v rozdelení jednotlivých stavov na viac podstavov, v našom prípade je však tento model postačujúci. Pre definované stavy si teda experimentálne vytvoríme maticu prechodov (tabuľka 5.2).

		<i>Budúci stav</i>										
		IF	KZ	AP	SP	P	N	I	T	O	PZ	U
<i>Súč. stav</i>	IF	0,05	0,95	0	0	0	0	0	0	0	0	0
	KZ	0	0,7	0,3	0	0	0	0	0	0	0	0
	AP	0	0	0,6	0,4	0	0	0	0	0	0	0
	SP	0	0	0	0,65	0,35	0	0	0	0	0	0
	P	0	0	0	0	0,55	0,45	0	0	0	0	0
	N	0	0	0	0	0	0,9	0,1	0	0	0	0
	I	0	0	0	0	0	0	0,97	0,03	0	0	0
	T	0	0	0	0	0	0	0	0,85	0,15	0	0
	O	0	0	0	0	0	0	0	0	0,8	0,2	0
	PZ	0	0	0	0	0	0	0	0	0	0,2	0,8
	U	0	0	0	0	0	0	0	0	0	0	1

Tabuľka 5.2: Matica prechodov pre model IT projektu

Na uvedenej matici prechodov možno pozorovať sekvenčný priebeh jednotlivých etáp projektu. Napr. etapa implementácie musí spravidla nasledovať po etape návrhu a podobne. Uvedené pravdepodobnosti prechodov sú dimenzované na trojdňový krok. Vidíme teda, že napr. inicializačná etapa na 95% prebehne už počas prvých troch dní trvania projektu. Zadanú maticu prechodov dáme na vstup našej implementovanej aplikácie a vytvoríme graf podľa vzťahu pre absolútne pravdepodobnosti zotrvania v jednotlivých stavoch systému (obrázok 5.10).

Získaný výsledok teraz porovnáme s dobou 270 dní, kedy je termín ukončenia projektu. V grafe vidíme, že v čase 270 dní (90 krokov) je 80% pravdepodobnosť, že projekt bude ukončený. V tomto momente je na manažérovi rizík, ako sa k danému výsledku postaví a aké opatrenie zvolí (v kapitole sú popísané základné typy spracovania rizík). Určitý priestor pre opatrenia v tomto modeli si vyžaduje najmä etapa implementácie, kde podľa grafu hrozí najväčší časový problém. Vytvorený model má už v kontexte analýzy rizík značný význam a môže slúžiť ako odrazový mostík k podobným modelom v rámci manažmentu rizík, čo možno považovať za nezanedbateľný prínos tejto diplomovej práce. Model je priložený na DVD s aplikáciou.



Obrázok 5.10: Grafická reprezentácia výpočtu pre 100 krokov

Markovova analýza je teda úspešne aplikovateľná nie len na menšie subsystemy v rámci projektu ako sa na začiatku predpokladalo ale aj na projekt ako celok. Uvedené modely sú toho jasným dôkazom. Manažérovi rizík, resp. akémukoľvek užívateľovi aplikácie, ktorá bola implementovaná v rámci tejto diplomovej práce už nič nebráni, aby mohol efektívne modelovať a vyhodnocovať Markovove modely v kontexte manažmentu rizík a manažmentu projektov všeobecne.

Kapitola 6

Záver

Cieľom tejto diplomovej práce bolo zoznámenie sa s problematikou manažmentu projektov a v rámci neho hlavne s pojmom manažment rizík. Jedná sa o pomerné rozsiahlu oblasť s niekoľkými etapami, kde sa najväčší dôraz kládol najmä na etapu analýzy rizík. V rámci analýzy rizík som sa zoznámil hneď s niekoľkými metódami a technikami, pričom som sa zamerával najmä na Markovovu analýzu a analýzu pomocou Petriho sietí.

Nadobudnuté vedomosti o danej problematike som použil pri návrhu a implementácii desktopovej aplikácie, ktorá slúži ako podpora pre manažérov rizík v rámci analýzy rizík v projektoch. Aplikácia implementuje Markovovu analýzu a jej funkčnosť je podrobne popísaná v tejto práci.

Markovova analýza (ako aj analýza pomocou Petriho sietí) je v praxi najčastejšie spájaná s modelovaním systémov technologického rázu. V kontexte manažmentu projektov sa jedná o pomerne veľkú neznámu, preto ďalším cieľom tejto práce bola práve demonštrácia tejto metódy v manažmente rizík IT projektov i projektov všeobecne.

Tento cieľ bol úspešne dosiahnutý analytickým vyhodnotením vytvoreného modelu, ktorý modeluje etapy IT projektu. Analýza prebehla v prostredí implementovanej aplikácie a uvedený model sa takto môže stať odrazovým mostíkom pre mnohých manažérov rizík, ktorí potrebujú efektívnu kvantitatívnu metódu vyhodnotenia stavov projektu.

V závere práce môžem konštatovať, že manažment rizík je bezpochyby dôležitou súčasťou manažmentu projektov a znižuje pravdepodobnosť ich neúspešného ukončenia. Markovova analýza sa ukázala ako veľmi efektívna metóda predikcie vývoja projektov a analýzy možných rizík, ktoré sú s nimi spojené.

6.1 Ďalší rozvoj aplikácie

V prípade budúceho rozšírenia aplikácie sa črtá niekoľko smerov, kam by sa mohla uberať. Súčasná aplikácia slúži len na analýzu rizík, pričom ich identifikácia a vyhodnotenie už je výlučne zodpovednosťou manažéra rizík. Preto by bolo v budúcnosti možné odraziť sa od tohto faktu a implementovať napr. tzv. register rizík, ktorý by bol spojený s konkrétnym projektom. V takom prípade by užívateľ mal možnosť prehľadného zobrazenia všetkých identifikovaných rizík v rámci projektu, čo by tiež výrazne uľahčovalo ich vyhodnotenie a rozhodnutia v prípade protirizikových opatrení. Aplikácia tiež momentálne funguje na báze vyplňania tzv. matice prechodov a graficky znázornený diagram stavov je len akýmsi doplnkom. Diagram stavov však skrýva veľký potenciál vďaka svojej prehľadnosti, preto je v budúcnosti vhodné rozšírenie o grafický editor Markovovho modelu.

Literatúra

- [1] dom4j. [online], [rev. 201-05-22], [cit. 2011-05-22].
URL <http://www.dom4j.org/dom4j-1.6.1/>
- [2] Fault Tree Analysis Software - Reliability Workbench from Isograph. [online], [rev. 2011-04-15], [cit. 2011-05-21].
URL <http://www.isograph-software.com/ftpover.htm>
- [3] JFreeChart. [online], [rev. 2009-08-08], [cit. 2011-05-22].
URL <http://www.jfree.org/jfreechart/>
- [4] NetBeans. [online], [rev. 201-05-22], [cit. 2011-05-22].
URL <http://netbeans.org/>
- [5] Vose Software / Vose Risk Consulting. [online], [rev. 2011-05-21], [cit. 2011-05-21].
URL <http://www.vosesoftware.com/>
- [6] COOPER, D. F.; GREY, S.; RAYMOND, G.; aj.: *Project Risk Management Guidelines: Managing Risk in Large Projects and Complex Procurements*. John Wiley and Sons, 2004, ISBN 0-470-02281-7, 400 s.
- [7] DOLANSKÝ, V.; MĚKOTA, V.; NĚMEC, V.: *Projektový manažment*. Grada, 1996.
- [8] DOLEŽAL, J.: *Predikce v projektech s využitím Markovských řetězců*. Dizertačná práca, Vysoké Učení Technické v Brně - Fakulta strojního inženýrství, 2010.
- [9] FUQUA, N.: Markov Analysis. [online], [rev. 2011-05-20], [cit. 2011-05-20].
URL <http://www.theriac.org/DeskReference/viewDocument.php?id=95>
- [10] HALANOVA, Z.: Manažment projektov: teória vs. prax. [online], [rev. 2005-06-25], [cit. 2011-05-05].
URL <http://www2.fiit.stuba.sk/~bielik/courses/msi-slov/kniha/2005/group3/essay/17.pdf>
- [11] HORTON, I.: *Java 5*. Neocortex, spol. s.r.o., 2005, ISBN 80-86330-12-5, 1443 s.
- [12] *ČSN EN 61165, Použití Markovových technik*. 2007, 32 s.
- [13] *ISO/IEC 31010, Risk management – Risk assessment techniques*. 2009, ISBN 2-8318-1068-2, 188 s.
- [14] *ISO 31000, Risk management – Principles and guidelines*. 2009, 24 s.
- [15] *ISO/IEC 15909 – 1, Software and systems engineering – High-level Petri nets*. 2004, 38 s.

- [16] KIMLIČKA, Š.: Projektový manažment – cesta k úspešnej elektronizácii knižnice. [online], [rev. 2007-12-21], [cit. 2011-05-05].
URL <http://www.inforum.cz/archiv/inforum1999/prednasky/kimlicka.htm>
- [17] PERINGER, P.: *Modelování a simulace – Studijní opora*. 2008, 85 s.
- [18] Pravdepodobnostné A Rozhodovacie Stromy. [online], [rev. 2011-01-10], [cit. 2011-01-10].
URL <http://www.euroekonom.sk/ekonomia/riziko/pravdepodobnostne-a-rozhodovacie-stromy/>
- [19] TAYLOR, J.: *A Survival Guide for Project Managers*. AMACOM Books, 1998, ISBN 0-814-403-379, 246 s.
- [20] VERNER, J. M.; EVANCO, W. M.: In-House Software Development: What Project Management Practices Lead to Success? 2005, iEEE Software, Vol. 22, No. 1, s. 86 – 93.
- [21] VOSE, D.: *Risk Analysis: A Quantitative Guide*. John Wiley and Sons, třetí vydání, 2008, ISBN 0-470-51284-9, 752 s.
- [22] ZENDULKA, J.; BARTÍK, V.; KVĚTOŇOVÁ, Š.: *Analýza a návrh informačních systémů – Studijní opora*. 2006, 178 s.
- [23] ČEŠKA, M.; MAREK, V.; NOVOSAD, P.; aj.: *Petriho sítě – Studijní opora*. 2009, 244 s.

Dodatek A

Obsah DVD

- **MarkovAnalysis**
Zložka s aplikáciou, spustiteľná ako projekt v prostredí NetBeans.
- **MarkovAnalysss SourceCodes**
Zložka so zdrojovými súbormi aplikácie
- **MarkovAnalysis Runnable**
Zložka so skompilovanou a spustiteľnou aplikáciou
- **Model Examples**
Zložka s príkladmi Markovových modelov, ktoré možno načítať v implementovanej aplikácii
- **JFreeCHart**
Zložka s knižnicami JFreeChart a JCommon spolu s licenčnou zmluvou
- **Technical report**
Elektronická verzia technickej správy diplomovej práce spolu so zdrojovými kódmi v \LaTeX .