

Prehlásenie autora

Čestne prehlasujem, že táto bakalárska práca je mojim pôvodným autorským dielom, na ktorom som pracovala samostatne na základe vlastných teoretických a praktických poznatkov získaných počas štúdia a informácií z dostupnej literatúry. Všetky zdroje, literatúru a pramene, ktoré som pri vypracovaní použila, riadne citujem s uvedením plného odkazu na príslušný zdroj. Uvedenú prácu som vypracovala pod vedením prof. RNDr. Gabriela Juhása, PhD..

Bratislava, dňa 20. 05. 2016

.....

Anna Demeterová

Pod'akovanie

Ďakujem rodine, priateľovi Matúšovi, vedúcemu práce prof. RNDr. Gabrielovi Juhásovi, PhD. a ďalším študentom, ktorí sa podieľali na tvorbe komplexného Workflow manažment systému, ktorého je táto práca súčasťou.

Abstrakt

Univerzita: Slovenská technická univerzita v Bratislave

Fakulta: Fakulta elektrotechniky a informatiky

Študijný program: Aplikovaná informatika

Školiace pracovisko: Ústav informatiky a matematiky

Autor: Anna Demeterová

Názov práce: Workflow manažment systém - procesný server

Vedúci práce: prof. RNDr. Gabriel Juhás, PhD.

Bratislava 2016

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Kľúčové slová: Petriho siete, workflow manažment, prípady použitia

Abstract

University: Slovak Technical University in Bratislava

Faculty: Faculty of Electrical Engineering and Information Technology

Study programme: Applied Informatics

Training workplace: Institute of Computer Science and Mathematics

Author: Anna Demeterová

Title: Workflow management system - process server

Thesis supervisor: prof. RNDr. Gabriel Juhás, PhD.

Bratislava 2016

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Keywords: Petri nets, workflow management, case

Obsah

0	Úvod	7
1	Analýza problému	8
1.1	Workflow manažment systémy	8
1.1.1	Case - prípad použitia	8
1.1.2	Task - úloha	9
1.1.3	Workflow rozmery	9
1.2	Petriho siete	9
1.2.1	História	9
1.2.2	Klasické Petriho siete	9
1.2.3	Spustiteľnosť a spustenie prechodu	10
1.2.4	Grafické znázornenie	11
1.2.5	Inhibítorové a reset hrany	11
1.3	WFMS založené na Petriho sieťach	11
1.3.1	Modelovanie smerovania	12
2	Analýza problému	14
3	Záver	15

Zoznam obrázkov

1	Grafické značenie objektov Petriho sietí	11
2	Grafické značenie inhibítor a reset hrán	11
3	Stavebné bloky modelovania workflow	12
4	Sekvenčné smerovanie	12
5	Paralelné smerovanie	13
6	Podmienkové smerovanie	13

0 Úvod

V praxi sa často stretávame s potrebou riadenia interných biznis procesov firiem alebo pracovných tímov. Väčšinou sa riadenie realizuje pomocou rôznych softvérových systémov. V týchto prípadoch nastáva problém nejednotnosti systémov, alebo potreby znalostí komplikovaného modelovacieho, prípadne programovacieho jazyka. Často je potrebné funkcionality nových procesov implementovať do zabehnutých systémov, čo je spojené so zvýšenými finančnými výdavkami a časovou investíciou.

Riešením tohto problému by bol jednotný systém, ktorý by umožňoval pomocou jednoduchého a ľahko pochopiteľného modelovacieho jazyka pridávať funkcionality rôznych procesov bez nutnosti odborného programovania.

Našou snahou bolo navrhnúť a vytvoriť takýto Workflow manažment systém, v ktorom sa procesy budú modelovať pomocou Petriho sietí a v jednoduchých editoroch na front-ende bude možné pridávať dodatočnú funkcionality, a to pridávanie formulárov a zdefinovanie používateľských rolí pre jednotlivé úlohy. Navyše, v systéme sa procesy budú môcť zdieľať medzi rôznymi firmami, čiže proces jednej firmy si budú môcť zakúpiť iné.

Cieľom tejto bakalárskej práce je namodelovaný proces pomocou Petriho siete uložiť do databázy, umožniť vytvorenie tzv. case-u a na základe jeho logiky ho reprezentovať vo forme úloh pre používateľov. Taktiež umožňuje vizualizáciu case-ov, aby bolo zistiteľné, v akom stave sa ktorý case nachádza, ktoré úlohy sú splnené, ktoré sa vykonávajú a ktoré ešte vykonané neboli.

1 Analýza problému

1.1 Workflow manažment systémy

Cieľom mnohých informačných systémov v dnešnej dobe je podpora a monitorovanie biznis procesov. Je žiadané, aby tieto systémy kontrolovali správny tok práce v organizáciách. Na základe tejto potreby a špecifikácie sa podľa[1] zrodil termín **Workflow manažment systém** (ďalej WFMS). W.M.P. van der Aalst v [1] definuje WFMS ako všeobecný softvérový nástroj, ktorý umožňuje definovanie, realizáciu, registráciu a kontrolu pracovných postupov. Termín workflow sa často používa ako synonymum pre biznis proces [9] a môžeme ho chápať ako rozdelenie komplikovanejšieho procesu na postupnosť jednoduchších krokov. Túto postupnosť je potrebné mať správne zadefinovanú, ak chceme, aby WFMS pracoval správne, ako je žiadané, pretože kvalita WFMS stojí na tomto zadefinovaní. WFMS ponúkajú riešenie pre podporu a optimalizáciu komplexnejších činností - procesov, riadi, aby aktivity boli plnené správnym človekom v riadnom čase v správnom poradí. Podľa Workflow Management Coalition[1] sa tieto systémy vzťahujú k doménam sústredeným na logistiku biznis procesov. *“Workflow manažment systémy úplne definujú, riadia a vykonávajú workflow pomocou spúšťania softvéru, ktorého poradie jednotlivých krokov je riadené počítačovou reprezentáciou workflow logiky.”* Tieto systémy nie sú limitované na špecifickú biznis aplikáciu, sú široko použiteľné.[9]

Ako príklad uveďme proces Žiadosť o pôžičku. Tento proces má pevne zadefinované kroky, ich postupnosť, pravidlá pre ich splnenie. Banka opakovane používa tento proces pre správny postup pri posudzovaní žiadostí.

1.1.1 Case - prípad použitia

Podľa [1] WFMS sú *založené na case-och*, čo znamená, že z procesov sú používateľmi vytvárané prípady použitia, takzvané case-y. Case je unikátna reprezentácia workflow procesu a má svoju identitu.[4] Cieľom WFMS je riadiť case-y čo najefektívnejšie. Opäť uvažujme proces Žiadosť o pôžičku. Zákazník požiadala o pôžičku vo firme používajúcej tento proces, považujeme to za jeden case, alebo prípad použitia. Ak o pôžičku zažiada iný zákazník, považujeme to za iný case. Taktiež ak by ten istý zákazník opätovne podal ďalšiu žiadosť, opäť je to nový case.

1.1.2 Task - úloha

Podľa [9] case-y procesov sú reprezentované ako tasky - úlohy. Sú to nedeliteľné - atomické logické jednotky procesu.[3] Tieto úlohy plnia používatelia, zamestnanci vo firme, prípadne tvorcovia case-u, čo bývajú často ľudia mimo firmu, ako napríklad zákazníci. S týmito úlohami môžu byť späté rôzne podúlohy, ako napríklad vyplnenie formulárov, dodanie dokumentov, a iné. V [1] je task, ktorý treba vykonať pre špecifický case definovaný ako **work item - pracovná položka**. Task, ktorý je vykonávaný, podľa tej istej literatúry nazývame **activity - aktivita**. Jednotlivé tasky sú vykonávané zdrojmi, ako napríklad, počítače, alebo častejšie ľudia. Ak opäť ako príklad použijeme proces Žiadosť o pôžičku, jednotlivé tasky by mohli byť napríklad Posúdenie žiadosti, Dohodnutie splátok, Podpísanie zmluvy a iné. Na úlohu Dohodnutie splátok môže byť naviazaná podmienka vyplnenia splátkového kalendára a jeho podpísanie žiadateľom.

1.1.3 Workflow rozmery

Na základe 1.1.1 , 1.1.2 , [1] a [9] má workflow 3 rozmery, a to:

- *Rozmer case-u* - znamená, že každý case je riadený individuálne. Case-y sa navzájom neovplyvňujú.
- *Rozmer procesu* - smerovanie procesu je špecifikované.
- *Rozmer zdrojov* - zdroje spracúvajú cas-y, vykonávajú úlohy.

1.2 Petriho siete

1.2.1 História

Pôvod Petriho sietí sa v [5] datuje v šesťdesiatych rokoch v skorej práci Carla Adama Petriho(12.júl 1926 – 2.júl 2010)[10]. Niektoré zdroje [6] uvádzajú, že Petri tieto siete vymyslel už v roku 1939 na modelovanie chemických procesov. Podľa [5], jeho skorá práca neobsahuje Petriho siete a ich definície tak, ako ich poznáme v dnešnej dobe. Doteraz existuje viacero definícií Petriho sietí.

1.2.2 Klasické Petriho siete

Podľa [2] klasické Petriho siete sú zvláštny druh orientovaného grafu. Tento graf pozostáva z 2 typov elementov, a to z **uzlov** a **hrán**.

Petriho sieť považujeme za graf **bipartitný** [1] [3], čo znamená, že uzly v týchto grafoch sú 2 typov, a to **miesto** - *place* a **prechod** - *transition*. Tieto uzly sú spojené pomocou **hrán** - *arcs*. Platí pravidlo, že 2 uzly toho istého typu nemôžu byť spojené hranou. [3] Ako sme spomínali v 1.2.1, definíci Petriho siete je viac, uvádzame definíciu z [7].

Definícia 1. *Petriho sieť je orientovaný bipartitný graf reprezentovaný štvoricou (P, T, F, W, m_0) , kde:*

$P = \{p_1, \dots, p_n\}$ - konečná neprázdna množina miest;

$T = \{t_1, \dots, t_n\}$ - konečná neprázdna množina prechodov ;

$P \cap T = \emptyset, P \cup T \neq \emptyset$;

$F \subseteq (P \times T) \cup (T \times P)$ - množina hrán (toková relácia);

$W: F \rightarrow N \cup \{0\}$ - váhová funkcia;

$m_0: P \rightarrow N \cup \{0\}$ - počiatočné značkovanie;

Váhová funkcia predstavuje množinu funkčných hodnôt váh všetkých hrán v Petriho sieti. Počiatočné značkovanie vyjadruje počiatočný stav Petriho siete, priradzuje každému miestu počet **značiek** - *tokenov*. [7] Ich počet v miestach sa počas spúšťania prechodov siete mení.

Miesta môžeme označovať ako vstupné a výstupné vzhľadom na jeden prechod. Ako vstupné ho označíme, ak hrana z neho vychádza a smeruje do daného prechodu. Naopak, z tohto prechodu hrana smeruje do miesta, ktoré označujeme výstupné.

1.2.3 Spustiteľnosť a spustenie prechodu

Prechod značíme **spustiteľný**, ak platí, že značkovanie každého vstupného miesta vzhľadom na daný prechod, je väčšie alebo rovné, ako váhová funkcia hrany, ktorá do skúmaného prechodu vchádza. Teda:

$$\forall p \in P : m(p) \geq I(p, t) \quad (1)$$

kde I je vstupná - *input* matica danej Petriho siete. [7]

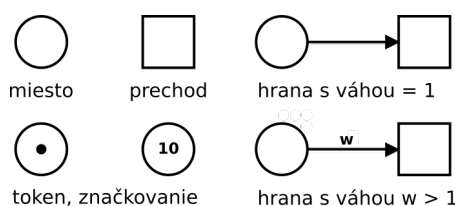
Ak prechod spĺňa tieto podmienky, jeho spustenie môžeme rozdeliť do 2 krokov: skonzumovanie tokenov a produkcia tokenov. Prechod spustíme a ten z každého vstupného miesta *skonzumuje* taký počet tokenov, aká je váha príslušných spojovacích hrán. Následne sa vyprodukujú tokeny do výstupných miest, pre každé miesto toľko, akú váhu má spojovacia hrana. Nastala zmena značkovania:

$$\forall p \in P : m'(p) = m(p) + O(p, t) - I(p, t) \quad (2)$$

kde O je výstupná - *output* matica danej Petriho siete. [7]

1.2.4 Grafické znázornenie

Petriho siete sú grafickým jazykom. [1] Miesta sú reprezentované kružnicou, prechody štvorcami a hrany oblúkmi, pričom šípka hrany indikuje smer toku. Váhy hrán sa značia číslom na hrany, ak je váha hrany rovná jednej, váha sa neznačí. Tokeny sa v grafoch značia ako čierne body v mieste, prípadne pri vyšších hodnotách ich značíme číselne.



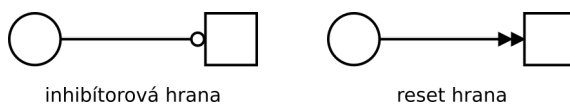
Obr. 1: Grafické značenie objektov Petriho sietí

1.2.5 Inhibítorové a reset hrany

V nasledujúcej podkapitole využívame informácie z [8]. V našom systéme využívame okrem normálnych hrán aj hrany inhibítorové a reset hrany. Jedná sa o jednoduché rozšírenie. Tieto hrany môžu byť použité na namodelovanie blokovania a vymazania. Obidva typy majú smer z miesta do prechodu, čiže sú to vstupné hrany.

Reset hrany umožňujú prechodu zkonsumovať všetky tokeny z miesta, z ktorého vychádzajú, čiže spustením prechodu v tomto mieste nezostane žiaden token. V sieti ich zaznačujeme dvojitou šípkou pri prechode.

Inhibítorové hrany povoľujú spustenie prechodu len v tom prípade, ak miesto z ktorého vychádza inhibítorová hrana obsahuje menší počet tokenov, ako je váha tejto hrany. Graficky sa tieto hrany neukončujú na prechode šípkou, ale malou kružnicou.



Obr. 2: Grafické značenie inhibítor a reset hrán

1.3 WFMS založené na Petriho sieťach

Petriho siete sú silný nástroj na špecifikovanie smerovania case-ov procesu. Tasky sú namodelované pomocou prechodov a závislosti sa modelujú pomocou miest,

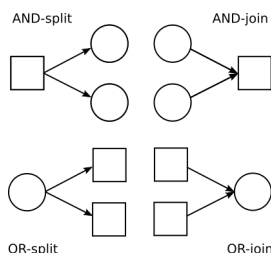
hrán a značkovania. Určuje dynamické správanie pre každý case z tohto procesu.[9] Petriho sieť, ktorá modeluje workflow proces sa nazýva WorkFlow sieť(d ďalej WF-sieť). [1]. WF siete majú 2 podmienky:

1. Jedno vstupné miesto (i) a jedno výstupné miesto (o). Token v i zodpovedá začiatku case - u a token v o case-u, ktorý už nemá tasky na splnenie, jeho prechody už nie sú spustiteľné a case je ukončený.
2. Každý task - prechod a podmienka - miesto prispievajú k spracovaniu case-ov, čiže všetky prechody a miesta by mali byť umiestnené na ceste z miesta i do miesta o [1]

Značkovanie WF-siete reprezentujú stav workflow-u jediného caseu.

1.3.1 Modelovanie smerovania

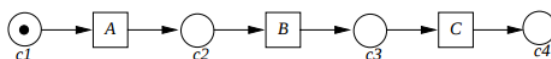
Na nasledovnom obrázku vidíme stavebné bloky, pomocou ktorých modelujeme logiku apodmienky WF procesov.



Obr. 3: Stavebné bloky modelovania workflow

Poznáme 4 typy smerovania

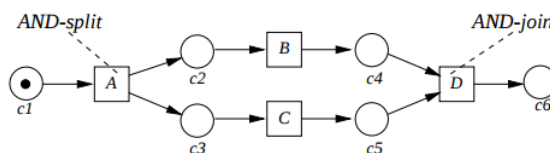
- *Sekvenčné* - rieši kauzálne vzťahy medzi úlohami, úlohy sa vykonávajú sekvenčne za sebou.



Obr. 4: Sekvenčné smerovanie

[1]

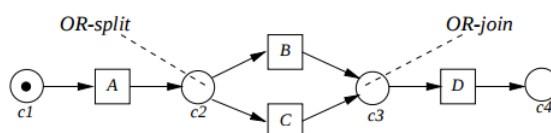
- *Paralelné* - využíva sa, keď sa niektoré úlohy môžu riešiť paralelne, prípade na poradí ich riešenia nezáleží. Na modelovanie paralelity sa používajú bloky AND-split a AND-join.



Obr. 5: Paralelné smerovanie

[1]

- *Podmienkové* - umožňuje rozhodovanie, smerovanie, v ktorom sa case-y môžu líšiť. Je závislé na atribútoch case-u. Toto smerovanie sa realizuje pomocou blokov OR-split a OR-join



Obr. 6: Podmienkové smerovanie

[1]

- *Iteračné* - pomáha riešiť situácie, ak prechod má byť spustený viackrát

[1][9]

2 Analýza problému

Úsek	dĺžka [m]	priemer. stúpanie [%]	priemer. rýchlosť [km/h]	reálny výkon [W]	Bike Calculator [W]	Rozdiel	Rozdiel [%]
1.	1000	7.4	13.5	269	247	-22	-8.18%
2.	1000	8.5	13.0	285	269	-16	-5.61%
3.	1000	10.2	11.9	285	289	4	1.40%
4.	1500	4.9	26.1	267	261	-6	-2.25%
5.	1500	6.6	15.6	272	263	-9	-3.31%
6.	1500	9.2	11.7	273	258	-15	-5.49%
7.	2700	3.0	20.4	201	197	-4	-1.99%

UC_01 Prihlásenie do systému

- Popis: prihlásenie používateľa do systému prostredníctvom jeho emailu a hesla
- Vstupné podmienky pre používateľa: správne heslo a email použitý pri registrácii
- Chybový tok: nesprávne vstupné údaje, používateľ môže akciu opakovať
- Zmena stavu systému: prihlásenie a presmerovanie na požadovanú alebo hlavnú stránku

3 Závěr

Literatúra

- [1] AALST, W.: *The Application of Petri Nets to Workflow Management*. Dostupné na internete: <http://www.wis.win.tue.nl/~wvdaalst/publications/p53.pdf>
- [2] DESEL, J. - REISIG, W.: *Lectures on Petri nets I: Basic models*. Germany: Universität Karlsruhe: Springer, 1998, ISBN 3-540-65306-6
- [3] LEWIS, A.H.: *An introduction to Petri nets*[cit.2016.04.25]. Dostupné na internete: <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=B8F8CDAA7484532525753>
- [4] EHRIG, H. a kol. *Petri Net Technology for Communication-Based Systems, Advances in Petri Nets*. Germany: Springer, 2003, ISBN 3-540-20538-1
- [5] EHRIG, H a kol. *Unifying Petri Nets*. Germany: Springer, 2001, ISBN 3-540-43067-9
- [6] PETRI, C.A. - REISIG, W. 2008 *Petri net*. [online] [cit. 2016.04.27]. Dostupné na internete: http://www.scholarpedia.org/article/Petri_net
- [7] JUHÁS, G.: *Modelovanie formalizmy udalostných systémov* Bratislava: RT systems s.r.o., 2011, ISBN 978-80-970519-1-4, p. 29-45
- [8] VERBEEK, H.M.W. a kol.: *Reduction Rules for Reset/Inhibitor Nets* Netherlands: Elsevier, 2009, Dostupné na internete: <http://www.wis.win.tue.nl/~wvdaalst/publications/p572.pdf>
- [9] AALST, W. - HEE, K.: *Workflow Management Models, methods and systems* Cambridge, Massachusetts: The MIT Press, 2004, ISBN 0-262-01189-1, p. 267-268
- [10] Prof. Dr. Carl Adam Petri [online] [cit. 2016.04.27] Dostupné na internete: http://www.informatik.uni-hamburg.de/TGI/mitarbeiter/profs/petri_eng.html

Obsah CD nosiča

Obsah priloženého CD nosiča je nasledovný:

- v priečinku *dokument* sa nachádza text diplomovej práce vo formáte PDF
- v priečinku *aplikácia* sú zdrojové súbory výslednej funkčnej aplikácie, a tiež skript pre vytvorenie tabuliek v databáze
- v priečinku *dokumentácia* je dokumentácia zdrojového kódu vo formáte HTML obsahujúca popis jednotlivých tried a ich metód