

DaemonX

Case nástroj pro efektivní modelování a adaptabilitu komplexních aplikací

(návrh SW projektu)

Vedoucí: Martin Nečaský (necasky@ksi.mff.cuni.cz)
Irena Mlýnková (mlynkova@ksi.mff.cuni.cz)

Řešitelé: Martin Chytil
Karel Jakubec
Vladimír Kudelas
Peter Piják
Marek Polák

Motivace:

Současné aplikace již typicky netvoří jednoduchý monolitický software, ale obvykle se jedná o složitý systém jednodušších aplikačních komponent, z nichž každá je zodpovědná za určitou výkonnou část. Takový systém má mnoho výhod, ovšem přináší s sebou také mnoho problémů. V první řadě je třeba umožnit uživatelsky příjemný a efektivní návrh jeho datových i procesních komponent, kterých může být velké množství. A máme-li už takový systém namodelovaný, je nutné řešit problematiku jeho evoluce, tj. následných změn uživatelských požadavků.

V současnosti existuje velké množství nástrojů, které umožňují realizovat první část problematiky, tj. modelování různých částí systému, jako je ER model [3] pro uložení dat např. do relační databáze, UML [4] model popisující např. objekty systému, model XSEM [5], popisující např. strukturu XML [1] rozhraní webových služeb nebo BPMN [2] model, kterým specifikujeme jejich výkonnou část. Málokdy se ovšem stane, že je první návrh, resp. realizace takového systému finální. Typicky se požadavky různých uživatelů na jednotlivé části systému mění nebo rozšiřují a taková změna může vynutit změnu většího množství komponent systému. Např. evidujeme-li v systému jméno zákazníka ve formě jednoduchého řetězce a přijde-li požadavek na strukturování této položky na jméno, příjmení a tituly, je nutné tuto změnu provést ve všech částech systému, které s touto informací pracují, tj. např. v úložišti dat, ve všech rozhráních webových služeb, které si tuto informaci předávají a ve všech postupech jejího zpracování. Potřebujeme tedy experta, který je schopen všechny dotčené komponenty identifikovat a určit zda a jaká změna má být provedena. V jednodušších systémech to není problém. Uvažujeme-li ale komplexní systém, kde jsou desítky nebo stovky rozhraní, objektových modelů, datových úložišť apod., jedná se o obtížný úkol, který není v silách jednotlivce a jehož manuální zpracování je náchylné k chybám.

Cíle projektu:

Cílem projektu je vytvoření case nástroje, který umožní návrhářům popsat změny uživatelských požadavků a ty semi-automaticky propagovat do schémat popisujících jednotlivé komponenty v systému, které jsou samy o sobě namodelovány v pro ně vhodných modelovacích jazycích. Konkrétně umožní návrhářům specifikovat danou změnu v podobě sekvence editačních operací nad schématem vybrané komponenty a odtud, pomocí vazeb mezi schémata komponent, ji propagovat do schémat ostatních komponent v systému. Case nástroj umožní i samotné modelování, ale to

není hlavním cílem projektu, neboť, jak bylo řečeno v úvodu, to již umožňuje celá řada existujících produktů.

Vzhledem k tomu, že se jedná o náročný úkol, budeme uvažovat v současné době nejrozšířenější datové modely, tj. UML, XSEM a ER, a procesní model BPMN, s důrazem na maximální možnou rozšiřitelnost o případné další modely podobného typu.

To, že je cílený problém adaptability netriviální demonstruje článek vedoucích projektu zabývající se adaptabilitou sady XML schémat vzhledem k měnícím se uživatelským požadavkům. Článek v současnosti prochází recenzním řízením a je možné ho interně poskytnout komisi k prozkoumání.

Hlavní cíle projektu tedy jsou:

- Návrh a implementace case nástroje, který umožní:
 - udržovat schémata jednotlivých komponent v systému popsaných ve vhodných modelovacích jazycích (prozatím UML diagramy tříd, XSEM, ER a BPMN),
 - udržovat vazby mezi schématy popisující sémantickou ekvivalenci mezi prvky schémat jednotlivých komponent (např. vazby mezi konceptuálním schématem domény, jednotlivými XML schématy a ER schématem)
 - editovat udržovaná schémata
 - propagovat editační operace mezi jednotlivými schématy prostřednictvím vazeb
 - rozšiřovat o další modely podobného typu, např. formou pluginů
- implementace ukázkové sady jazyků pro modelování datových schémat (např. UML diagramy tříd, XSEM a ER), které umožní demonstrovat propagaci změn mezi schématy modelovanými v odlišných jazycích
- implementace jednoho jazyka pro modelování procesů (např. BPMN), který umožní demonstrovat propagaci změn mezi datovým a procesním modelem, tedy zcela odlišnými pohledy na aplikaci
- Systém, který bude z pohledu uživatele umožňovat:
 - souběžné modelování více komponent
 - souběžné modelování komponenty pomocí různých modelovacích jazyků (např. pro modelování na různých úrovních abstrakce – konceptuální a logická)
 - souběžné modelování datové i procesní části aplikace
 - určení vazeb mezi schématy souvisejících komponent a mezi schématy popisujícími různou úroveň abstrakce jedné komponenty
 - editaci všech schémat pomocí atomických i složených operací, včetně funkce undo
 - podporu adaptability systému, tj. propagaci změn z jednoho schématu do všech navázaných schémat
 - uživatelskou interakci při propagaci změn

Další požadavky na program:

- Program by měl být řešen jako freeware aplikace, jejíž instalace nebude vyžadovat složité úkony, bude pokud možno přenositelná atd. Cílem je zajistit, aby aplikaci využívalo co nejvíce uživatelů.
- Veškerá dokumentace bude v angličtině
- K projektu vznikne odpovídající webová stránka, která jej bude detailně popisovat.

Předpokládaný průběh práce:

1. Analýza existujících implementací a přístupů – viz např. [6 – 12]
2. Podrobná specifikace konkrétních funkcí systému, architektury a rozhraní mezi jednotlivými moduly
3. Implementace projektu
4. Testy na reálných schématech, ladění
5. Dokumentace (programátorská, uživatelská, instalační)

Poznámka:

Problematiku řešenou v rámci implementace projektu je možné rozšířit do diplomových prací.

Doporučená literatura:

[1] Extensible Markup Language (XML) <http://www.w3.org/XML/>

[2] Business Process Model and Notation (BPMN) <http://www.omg.org/spec/BPMN/>

[3] Home page of Dr. Peter Chen: <http://bit.csc.lsu.edu/~chen/chen.html>

[4] UML® Resource Page <http://www.uml.org/>

[5] XSEM - A Conceptual Model for XML:
<http://crpit.com/abstracts/CRPITV67Necasky.html>

[6] XCase - Tool for XML Data Modeling: <http://xcase.codeplex.com/>

[7] Schema Evolution publication categorizer: <http://se-pubs.dbs.uni-leipzig.de/>

[8] Klettke, M.: Conceptual XML Schema Evolution - the CoDEX Approach for Design and Redesign. In BTW Workshops, pages 53-63. Verlagshaus Mainz, Aachen, 2007.

[9] Fiedler, G. - Thalheim, B.: An Approach to Conceptual Schema Evolution. Technical Report 0701, Institut für Informatik der Christian-Albrechts-Universität, Kiel, 2007.

[10] Su, H. - Kramer, D. K. - Rundensteiner, E. A.: XEM: XML Evolution Management. Technical Report WPI-CS-TR-02-09, Computer Science Department, Worcester Polytechnic Institute, Worcester, Massachusetts, 2002.

[11] Guerrini, G. - Mesiti, M. - Sorrenti, M. A.: XML Schema Evolution: Incremental Validation and Efficient Document Adaptation. In XSym '07, pages 92-106, Berlin, Heidelberg, 2007. Springer.

[12] Guerrini, G. - Mesiti, M.: XML Schema Evolution and Versioning: Current Approaches and Future Trends. Open and Novel Issues in XML Database Applications: Future Directions and Advanced Technologies, Idea Group Publishing, 2008/9.