

## Posudek oponenta diplomové práce Pavla Kadlece „Distribučování zpracování business procesů v prostředí podnikové sběrnice služeb“

Hlavním úkolem diplomanta bylo navrhnout prostředí, které umožní distribuované zpracování vnitropodnikových dávkových tzv. „business“ procesů a umožní jejich monitorování a řízení. Diplomová práce Pavla Kadlece všechny tyto body splňuje. V práci jsou dokumentovány všechny zmíněné součásti práce, systém mi byl předveden jako funkční (v simulované verzi). Podle vyjádření firmy Flynet uvedeném v závěru se zdá, že se osvědčil i v ostrém nasazení. Ocenit lze rovněž obrovské množství nástrojů a prostředí, která autor ve své práci použil.

Práce postupně popisuje jednotlivé problémy a diskutuje jejich řešení. Zde bych práci vyčetl trochu „dokumentaristický“ styl. Uvažme např. odstavec 5 „Volba middlewaru“. Zde se píše: „Máme na výběr mezi asynchronní komunikací a synchronní komunikací. Vzhledem k vytyčeným cílům je nejvhodnější použít posílání zpráv (messaging) jako komunikačního prostředku. Hohpe a Woolf uvádějí následující výhody messaging ...“. A již se mluví o výhodách a nevýhodách posílání zpráv, aniž by na začátku byl zmíněn princip této techniky, zda je synchronní či asynchronní. Od čtenáře se trochu předpokládá, že všechno zná.

K práci mám následující dotazy a poznámky:

- V práci se využívají modely procesů pořízené pomocí kreslicího modeláře Eclipse. Navrhne-li v modelu procesu paralelní větev, mohou být základem distribuce zpracování. Autor nevyužívá obecné možnosti pro rozvětvení (fork) a spojení cest (join), používá svá vlastní primitiva „spiliter“ a „aggregator“, které jsou ale uzpůsobeny řešení problematice. Tím se ale na druhé straně snižují možnosti ovlivnit při návrhu procesního modelu způsob distribuce.
- Proč nejsou využity diagramy aktivity UML, jejichž sémantika je zřejmá? Jedním z cílů práce (v odstavci 3.2) je možnost popsat „business proces“ pomocí diagramu aktivity a ten pak chápat jako „program“ pro „business stroj“, který jej interpretuje, citují: *„V ideálním případě by ... Spojení částí BP by již nebylo realizováno ručním kódováním, ale grafickým navrhováním BP pomocí nějakého CASE nástroje. Architekt v CASE nástroji navrhne plán BP – v řeči Unified Modeling Language (UML) vytvoří Activity diagram. Tento plán „nahraje“ do vytvořené infrastruktury. Tím je BP připraven k použití.“*
- Proč není v práci nikde uveden datový model? V odstavci 11.3.2 se hovoří o odstranění zbytečných indexů. Píše se „Na tabulce „transactions\_“ se nacházelo množství indexů – kde se ale vzaly? Někdo je přece musel vyrobit. Použitý databázový systém Oracle10g má prostředky pro sledování využití indexů (mimo jiné i např. SQL Developer zmíněný v použitých nástrojích).
- V odstavci 13.1 se konstatuje, že „začlenění více počítačů do procesu zpracování je z pohledu aplikačního programátora transparentní“. To ale může být někdy kontraproduktivní, neboť tento nemůže distribuci zátěže ovlivnit?

Přes výše uvedené výhrady považuji práci za velmi přínosnou a prakticky nesporně užitečnou. Vzhledem k objemu odvedené práce ji navrhuji rozhodně přijmout k obhajobě

V Praze 20. ledna 2009

Doc. Ing. Karel Richta, CSc.  
katedra softwarového inženýrství MFF UK Praha  
Malostranské nám.25, 118 00 Praha 1  
oponent