

# Vzťahy medzi členmi dodávateľského reťazca

PETER TRNOVSKÝ

*Slovenská technická univerzita  
Fakulta informatiky a informačných technológií  
Ilkovičova 3, 842 16 Bratislava*

**Abstrakt.** V posledných štyridsiatich rokoch sa techniky, metódy a procesy dodávky softvéru zameriavali najmä na dodávateľa softvéru. No s posunom požiadaviek trhu na čas, funkcionálnosť a flexibilitu, je potrebný posun aj v oblasti dodávky softvéru a to smerom k zákazníkovi. Evolúcia dodávateľských prístupov v softvérovom inžinierstve spôsobuje posun od dodávky softvéru ako tovaru, k dodávke softvérových služieb. Je zaujímavé sledovať vývoj dodávateľských vzťahov ako súčasť tejto evolúcie. Nemožno však zabúdať na široko rozšírený spôsob dodávky založený na využití generického softvéru, prípadne na vývoj softvéru na zákazku. Nové prístupy k dodávke softvéru, vytvárané s ohľadom na otvorený trh, prinášajú okrem zmeny organizácie dodávateľských vzťahov aj nové požiadavky na ich kvantitu. Súčasťou nových prístupov sú plne automatizované vzťahy pre koncového zákazníka ako v oblasti vývoja softvéru a komunikácie, tak aj v oblasti obchodu. Plne automatizované vzťahy znamenajú nielen prínos pre skvalitnenie vzťahov v procese dodávky softvéru, ale v konečnom dôsledku umožňujú zrýchliť dodávkový cyklus a evolúciu. Naopak klasické prístupy sú zaujímavé z pohľadu sily vzťahov a vzájomného poznania sa obchodných partnerov. Silný vzťah má stále svoje opodstatnenie vo vzájomnej dôvere, zdieľaní informácií, ktoré umožňujú znížiť riziko spojené s dodávkou softvéru.

## Vzťahy, dedičstvo minulosti

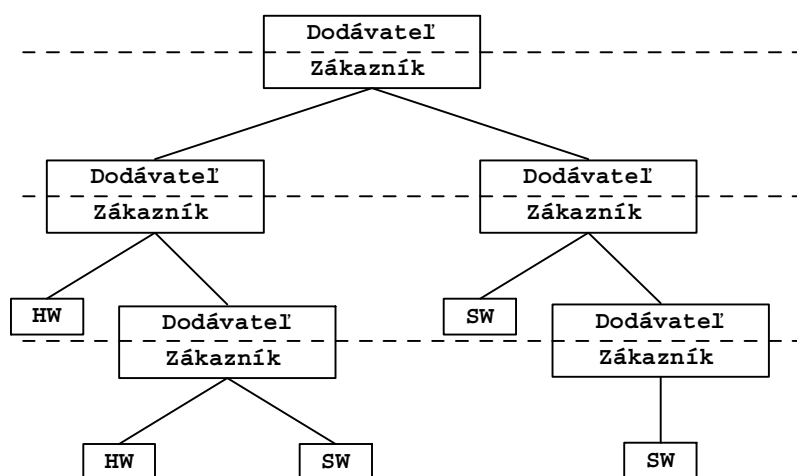
Dôležitosť a fungovanie vzťahov medzi zákazníkom a jeho dodávateľom sú veci, ktoré boli známe obchodníkom už pred niekoľkými stáročiami. Rovnaké vzťahy zostávajú platné dodnes a to aj pre oblasti, ktoré svoj vek počítajú v desiatkach rokov. Vzťahy medzi zákazníkom a dodávateľom majú svoje opodstatnenie aj v procese dodávky produktov a služieb v oblasti softvéru. Manažment vzťahov v moderných dodávateľských procesoch má rovnaké základy, ktoré vychádzajú z toku tovarov a služieb od dodávateľa k zákazníkovi [2, 4].

Organizácia zapojená do dodávateľského procesu si nevyhnutne vytvára vzťahy s dodávateľmi a zákazníkmi. Prepojením organizácií vznikajú špecifické dodávateľské

*Manažment v softvérovom inžinierstve, máj 2005, s. 1-8.*

reťazce. Do reťazca môže byť zapojené veľké množstvo organizácií. Každá z organizácií si vytvára vlastné vzťahy, ktorých počet a kvalita je vlastná danej organizácii. Vzájomné vzťahy hrajú dôležitú rolu v úspechu celého dodávateľského reťazca.

Klasický dodávateľský reťazec [5] je hierarchická organizačná štruktúra, ktorej vrcholom je koncový spotrebiteľ. Tento je prostredníctvom maloobchodného trhu napojený na sieť dodávateľov. Vzťahy medzi dodávateľmi sú organizované prostredníctvom veľkoobchodného trhu. Na úplnom spodku klasického dodávateľského reťazca sa nachádzajú primitívne výrobky a služby. Tokom primitívnych výrobkov ku koncovému zákazníkovi postupne narastá ich pridaná hodnota. Uvedený reťazec je možné nájsť aj v procese dodávky softvéru (obr. 1.).



Obr. 1. Prirodzený dodávateľský reťazec

Pri dodávke softvéru, ale aj v iných oblastiach, existujú vysoko dynamické, agilné organizácie s vysokou mierou inovácií, krátkym produkčným cyklom a prispôbovaním založenom na evolúcii zákaznických preferencií. U týchto je možné pozorovať klasickú dodávateľskú reťaz, ale táto je vzhľadom na rýchly vývoj, modifikovaná pri každom jej použití. Preto sa zameriame hlavne na podstatu vzťahov v sieti dodávateľov v závislosti od zvolenej vývojovej paradigmy.

V posledných štyridsiatich rokoch sa techniky, metódy a procesy dodávky softvéru zameriavali najmä na dodávateľa softvéru. No s posunom požiadaviek trhu kladených na funkcionality, flexibilitu a čas, je potrebný posun aj v oblasti vývoja a dodávky softvéru. Radikálny posun je potrebný najmä od centrického modelu založeného na požiadavkách, k modelu založenému na otvorenom trhu a softvérových službách.

## Otvorený trh

V absolútne flexibilnom svete má zákazník možnosť dostať službu kedykoľvek potrebuje jej vykonanie. Medzi problémom a jeho riešením existuje neskoré viazanie, ktoré sa vykonáva až v čase uspokojovania potreby. Predpokladá sa široká škála pre výber riešenia problému, jednoduchá nahraditeľnosť riešenia iným a bezprostredná dostupnosť.

Ideálnu predstavu trhu v praxi nahrádza pragmatickejšia predstava, v ktorej s využitím síl pôsobiacich na trhu vznikajú spojenectvá dodávateľov. Títo poskytujú menšie množstvo, ale populárnych riešení. Existuje tu teda tlak na ich neustále zlepšovanie. Tento stav je pozorovateľný vo viacerých odvetviach najmä v oblasti služieb. Na podpore otvoreného trhu je v súčasnosti v softvérovom inžinierstve založených viacero vývojových postupov (napríklad aj v rámci ESA). Dôvodom pre podporu otvoreného trhu je najmä tlak na rýchlu evolúciu riešení, či vytváranie rozsiahlych globálnych trhov.

## Softvér na zákazku

V súčasnosti bežným prístupom k dodávke softvéru je vytváranie softvéru na zákazku. Softvérom na zákazku budeme rozumieť vývoj softvéru pre koncovú organizáciu, kedy dodávateľská organizácia disponuje všetkou potrebnou technológiou, a teda jej fungovanie závisí len od malého množstva dodávateľov. Nosná časť vytváraného riešenia je dielom dodávateľa softvéru. Pre dodávku softvéru si zákaznícka organizácia vyberá jednu alebo malé množstvo dodávateľských organizácií.

Dodávka softvéru sa zväčša uskutočňuje dlhší čas, a preto je tento prístup charakteristický silným vzťahom medzi zákazníkom a dodávateľom. Vzťah medzi zákazníkom a dodávateľom sa spravidla vytvára v priebehu viacerých fáz. Fázy zahŕňajú stretnutia so zákazníkom za účelom špecifikácie požiadaviek, návrh systému, implementáciu, nasadenie, zaučenie používateľov, ladenie v čase prevádzky a následnú technickú podporu. Vzťah pri dodávke softvéru na zákazku preto pretrváva zväčša po dobu dodávky viacerých produktov alebo verzií softvéru.

Dobré poznanie sa oboch strán znižuje riziko neúspechu. Silný vzťah je výhodou aj pri analýze súčasných a budúcich potrieb zákazníka a znižuje riziko uviaznutia problému. Veľkou nevýhodou dodávky softvéru na zákazku je obmedzená konkurencieschopnosť v dobe po výbere dodávateľa. Pre zákazníka nie je jednoduché zmeniť dodávateľa alebo produkt v ďalších fázach dodávky.

Veľkým rizikom pre úspech projektu je nízka angažovanosť ľudí na strane zákazníka. Tento prístup je preto limitovaný na použitie najmä vo veľkých zákazníckych organizáciách, ktoré sú spôsobilé poskytnúť dostatočné množstvo a aktiváciu ľudskej sily.

Ďalšou nevýhodou je obmedzený prístup nových dodávateľov na trh. Prístup k trhu majú najmä silné dodávateľské organizácie. Hlavnou prednosťou týchto organizácií je ich meno, ktoré sa spája s vysokým štandardom kvality a garanciou úspešného ukončenia projektu.

## Generický softvér

Iným prístupom k dodávke softvéru, je vytváranie riešenia s využitím generického softvéru alebo COTS komponentov (Commercial off-the-shelf). Prístup COTS je založený na procese zostavovania konečného produktu, výberom z malého množstva nastavení. Stavebnou časťou pri prístupe COTS sú rozsiahlejšie produkty alebo softvérové celky, ktoré sú prispôbované na poskytovanie špecifickej funkcionality. Takéto softvérové celky často poskytujú a využívajú funkcionality v rámci značnej časti podnikových informačných systémov. Zákazníkmi pri dodávke týchto systémov sú priamo koncové organizácie.

Vývojom procesu pre vytváranie COTS systémov sa zaoberá množstvo vývojových skupín. Príkladom úspešných procesov sú procesy PORE [7] a PECA [3]. Súčasťou procesov sú aj postupy pre vytváranie vzťahov medzi dodávateľmi COTS produktov a organizáciami vyhľadávajúcimi COTS produkty.

Dodávka softvéru na zákazku a prístup COTS majú z hľadiska vzťahov dodávateľa a koncového zákazníka mnoho spoločného. Pri týchto prístupoch vstupuje koncový zákazník do dodávateľského vzťahu s jednou alebo malým množstvom dodávateľských organizácií. Medzi zákazníkom a dodávateľom vzniká silný vzťah. Z pohľadu oboch strán sa javí druhá strana ako dlhodobý partner. Medzi oboma stranami sa vytvára silnejší vzťah – partnerstvo. Aby bolo partnerstvo úspešné musia byť vzájomné vzťahy budované na silných základoch. Vzťahy sú budované zväčša dlhodobo a rozširujú sa s priebehom dodávky produktu alebo produktov. Vo vzťahoch je nevyhnutný vzájomný rešpekt a vôľa zdieľať informácie.

Ulohou zákazníka pri COTS prístupe je sledovanie technológie a udržiavanie informácií pre vyhodnocovanie a expertízy. Pri prístupe COTS je pre zákazníka žiaduce, aby bol schopný identifikovať a vyhodnotiť alternatívy a iných partnerov. Udržiavanie týchto informácií kompenzuje silnú viazanosť zákazníka na dodávaný produkt, prípadne na dodávateľskú organizáciu.

Dodávateľ sústavne konzultuje so zákazníkom jeho aktuálne a budúce požiadavky. Uprednostnením osobných kontaktov pred elektronickou komunikáciou si môže utvrdzovať vzťah so zákazníkom. Široká vzájomná komunikácia s partnerom, charakteristická pre tieto prístupy k dodávke softvéru, umožňuje predísť uviaznutiu problému či už na strane dodávateľa alebo zákazníka. Naopak veľké množstvo zdieľaných informácií nesie so sebou riziko straty potreby služieb dodávateľa. Preto pre správne fungovanie zdieľania informácií je dôležité, aby zákazník ubezpečoval dodávateľa, že mu nehrozí vytlačenie z trhu samotným zákazníkom.

Nevýhodou uvedených prístupov je vysoká viazanosť zákazníka na dodávateľa. Pre zákazníka môže byť veľmi ťažké zmeniť dodávateľa alebo nahradiť produkt. Pre malé a stredné organizácie môže byť ťažké dostať sa medzi dodávateľov produktu, prípadne stať zákazníkom pre tieto produkty.

Jednou z osobitných charakteristík dodávky softvéru na zákazku a COTS prístupu je možnosť manipulovať s druhou stranou. Táto možnosť je ako na strane zákazníka, tak na strane dodávateľa. Túto možnosť využívajú silné zákaznícke organizácie na vytváranie neprimeraného tlaku na svojich dodávateľov. Tieto organizácie sú však

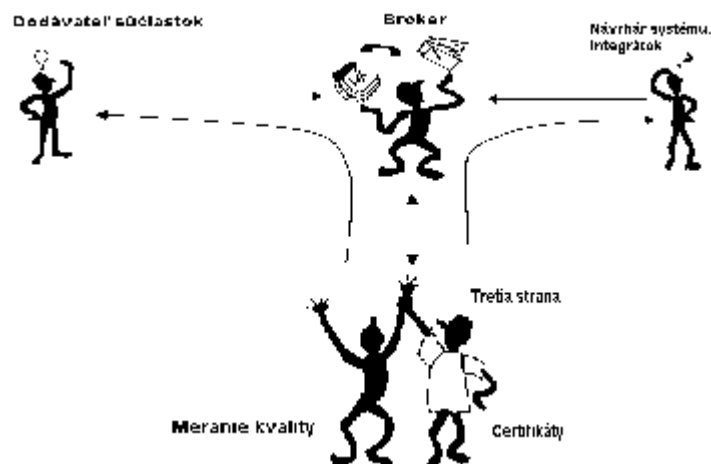
taktiež schopné poskytnúť vysoké nasadenie pracovníkov pre stretnutia s dodávateľom, či sledovanie demonštrácií produktu.

## Softvér zo súčiastok

Softvérové inžinierstvo založené na komponentoch CBSE (z anglického Component Based Software Engineering) je založené na vytváraní systémov z už existujúcich častí. V tejto eseji ním budeme rozumieť proces vytvárania systémov z komponentov, ktoré pochádzajú od nezávislých tvorcov, nezávislých dodávateľov a rozmiestnenia, pričom spolupracujú na vytvorení požadovanej funkcionality.

Dodávateľská reťaz v prístupe CBSE (obr. 2.) pozostáva najčastejšie zo štyroch skupín hráčov:

- Návrhár systému alebo integrátor komponentov – vyberá a získava komponenty. V CBSE prístupe predstavuje konečného zákazníka.
- Dodávateľ CBSE komponentov – vyvíja komponenty pre použitie v prístupe CBSE.
- Broker – vystupuje ako sprostredkovateľ. Vytvára automatizovaný trh pre dodávateľov a odberateľov CBSE komponentov.
- Tretia strana – vytvára pridanú hodnotu CBSE komponentov. Táto môže byť vo forme certifikátov alebo vyhodnocovania komponentov.



Obr. 2. Organizácia vzťahov pri prístupe CBSE

Vzťah medzi koncovým zákazníkom (tento raz návrhár systému) je spravidla realizovaný prostredníctvom sprostredkovateľa. Tento poskytuje množstvo služieb a najmä podporu pre rozhodovanie.

Dodávateľ súčiastok vstupuje do vzťahu so sprostredkovateľom s cieľom zaregistrovať svoju súčiastku v katalógu súčiastok, ktorý vedie sprostredkovateľ. Katalóg obsahuje funkcionálne a nefunkcionálne vlastnosti súčiastky a ďalšie informácie, ako napríklad obchodné informácie o dodávateľovi alebo cene. Zákazník si vyberá súčiastky z katalógu a v spolupráci so sprostredkovateľom robí kompromisy medzi ponukou súčiastok a vlastnými požiadavkami. Pri výbere mu pomáhajú nástroje pre usporiadanie a výber súčiastok, certifikáty, vizualizácia súčiastok a vizualizácia podobnosti súčiastok s požiadavkami. Vzťah medzi koncovým zákazníkom a dodávateľom súčiastok je preto takmer automatizovaný.

Nevýhodou CBSE prístupu je riziko spojené s trhom. Zákazník nemá taký silný vzťah so svojim dodávateľom ako v prípade dodávky softvéru na zákazku alebo COTS. Zvyšuje sa teda riziko, že dodávateľ prestane dodávať nájdenú súčiastku alebo úplne odíde z trhu. S tým je spojené riziko, že zákazník nebude vedieť nájsť náhradu súčiastky.

Posun od prístupu COTS nastal najmä v miere zviazanosti produktu so zákazníkom. Nahradenie softvérovej súčiastky v prístupe CBSE je principiálne jednoduchšie ako výmena rozsiahleho COTS komponentu.

Inou nevýhodou CBSE prístupu je vysoká réžia spojená s registráciou súčiastky. Táto časť by mohla byť v budúcnosti realizovaná na vyššej úrovni automatizácie.

## **Softvér ako služby**

SBSE – softvérové inžinierstvo založené na softvérových službách (z anglického Service Based Software Engineering) predstavuje evolučný krok z CBSE. Základom pre SBSE prístup je myšlienka zviazať a nastaviť služby na špecifickú množinu požiadaviek až v čase jej poskytnutia. Ako definíciu služby použijeme definíciu podľa [6] :

“Služba je pôsobenie alebo výkon ponúkaný jednou stranou druhej. Výkon môže byť zviazaný s fyzickým produktom. No v podstate je výkon nehmateľný a spravidla sa neprenáša do vlastníctva žiadneho faktoru produkcie.“

Dodávka služby môže pozostávať z poskytnutia základnej služby, alebo častejšie poskytnutia služby pozostávajúcej z čiastkových služieb. Za služby sa platí až priamo pri ich poskytnutí. Služba nie je mechanický proces, pretože zapája ľudský manažment dodávateľských vzťahov. Dodávka softvéru vo forme služieb umožňuje vytvárať, zhromažďovať a skladať služby spojením veľkého množstva dodávateľov, a to v dobe vzniku požiadavky na vykonanie služby [1].

Podobne ako v CBSE aj pri SBSE vystupujú tri typy hráčov:

- Zákazník – koncová organizácia, ktorá bude službu využívať.
- Dodávateľ SBSE služieb – na základe požiadaviek trhu vytvára a poskytuje služby koncovému zákazníkovi.
- Broker – vystupuje ako sprostredkovateľ. Vytvára automatizovaný trh pre dodávateľov a odberateľov SBSE služieb.

Rovnako ako pri prístupe CBSE, dodávatelia registrujú ponuku služieb v registri vedenom sprostredkovateľom. Zákazníci pristupujú k registru v dvoch krokoch. V prvom kroku zákazník prehľadáva služby s cieľom nájsť také, ktoré korešpondujú s jeho potrebami. V druhom kroku zákazník vybrané služby využíva. Vzťah medzi dodávateľom a zákazníkom je vytváraný dynamicky v čase poskytnutia služby. V súčasnosti je dostupné množstvo alternatívnych technológií, ktoré umožňujú podporu uvedeného modelu (.NET, J2EE a množstvo konkurenčných aplikačných serverov).

Počas návrhu služby, používateľ využíva prostredie pre návrh skladania služieb. Výsledkom je popis zloženej služby alebo šablóna. Návrh zloženej služby je časť procesu vyžadujúca si zapojenie ľudskej aktivity. Popis zloženej služby obsahuje architektúru čiastkových služieb, popis čiastkových služieb a podmienky na ich kvalitu.

V čase potreby vykonania služby je využitý systém dynamického poskytovania služieb. Tento na základe popisu alebo šablóny vykoná výber potrebných čiastkových služieb, automaticky dohodne podmienky, a to vrátane obchodných a zmluvných, s poskytovateľmi služieb a zavolá dohodnuté čiastkové služby. Spôsob viazania služieb sa približuje k ultra neskorému viazaniu, kedy sa zviazanie realizuje až pred poskytnutím služby a je zrušené bezprostredne po jej zrealizovaní. Zrejmou výhodou je jednoduchá nahraditeľnosť čiastkovej služby. Zákazník je minimálne viazaný na konkrétneho dodávateľa.

Ako bolo spomenuté, vzťah medzi zákazníkom a dodávateľom služby je realizovaný prostredníctvom sprostredkovateľa. Tento plní viacero úloh a môže pozostávať z viacerých organizácií. Tieto zabezpečujú registráciu služieb, platby za služby, dohadovanie zmluvných podmienok, monitorovanie kvality služieb a iné činnosti súvisiace s poskytnutím služieb.

Jednou z nevýhod uvedeného prístupu je riziko spojené s veľkou mierou automatizácie vzťahov. Dôvera medzi partnermi je ohraničená. Dôveru ohraničuje najmä slabé poznanie sa partnerov. S nízkou mierou dôvery je spojené riziko zlyhania. Inou nevýhodou, podobne ako pri CBSE prístupe, je vysoká réžia súvisiaca automatizáciou aktivít ako výber dodávateľa, dohadovanie zmluvných podmienok, zmena požiadaviek alebo monitorovanie kvality služieb.

## Záver

Jednoznačnou prednosťou poskytovania softvérových služieb pred ďalšími uvedenými prístupmi je priblíženie sa k otvorenému trhu. No každý z uvedených prístupov prináša do vzťahov svoje výhody a špecifiká. Na tieto nemožno zabúdať najmä pri evolúcii a vytváraní nových prístupov k tvorbe softvéru alebo modelov dodávateľských vzťahov.

Pri pohľade do budúcnosti je potrebné si uvedomiť výhody prístupu k dodávke softvéru vo forme služieb najmä v kontexte vytvárania globálnych trhov, dynamických vzťahov v sieti dodávateľov, nezávislého rozmiestnenie malých čiastkových služieb a umožnenie rýchlej evolúcie čiastkových služieb.

## Použitá literatúra

1. K. Bennett, P. Layzell, D. Budgen, P. Brereton, L. Macaulay, M. Munro: Service-Based Software: The Future for Flexible Software. APSEC, 2000.
2. P. Brereton: The Software Customer/Supplier Relationship. Communication of the ACM, Vol. 47, No. 2, February 2004, 77-81.
3. S. Comela-Dorda: A process for COTS software product evaluation. In Proceedings of the First International Conference on COTS-Based Software Systems, February 2002, Springer-Verlag.
4. R.B. Handfiels, E.L. Nichols: Introduction to Supply Chain Management. Prentice Hall, Upper Saddle River, NJ, 1998.
5. M. Jones, A. Mantineo, U.K. Mortensen: Introducing ECSS Software-Engineering Standards within ESA. Esa bulletin, 111., August 2002, 132-139.
6. C. Lovelock, S. Vandermerwe, B. Lewis: Services Marketing, Prentice Hall Europe, 1996.
7. N.A.M. Maiden, C. Ncube: Acquiring COTS software selection requirements. IEEE Software 15, 2, 1998, 45-56.

## Annotation

*Relationships between members of supply chain.*

Management of a supply chain is important step to implement more flexible and agile software development process. This paper present some approaches of software procurement. There are summarized advantages and future use of bespoke, component based and software services based software development processes. Major part of this work is dedicated to highlighting benefits of software services that lead to binding of service components just in time they are needed and then the binding may be discarded. Also this approach allows to make other relationships on demand.