

Programinės paslaugos modeliai ir jų ryšys su architektūriniais sprendimais

Marius Kaziliūnas

Vilniaus universiteto
Matematikos ir informatikos instituto doktorantas
Institute of Mathematics and Informatics,
Vilnius University, Doctoral student
Akademijos g. 4, LT-08663 Vilnius
Tel. (8 5) 210 9327
El. paštas: *marius.kaziliunas@mii.vu.lt*

Paslaugų paradigma yra teorinių ir metodinių prielaidų visuma, kuri kompiuterinių sistemų kūrimą nagrinėja kaip jų komponavimą iš programinių paslaugų. Tai besivystantis kompiuterinių sistemų kūrimo būdas, todėl egzistuoja didelė sąvokų sampratų įvairovė, laikomasi skirtingų požiūrių. Straipsnyje nagrinėjami paslaugų modeliai – analizuojamos paslaugų apibrėžtys, aptariamas ir lyginamas panašių elementų, priklausančių skirtingiems modeliams, turinys. Pateikiamas programinės paslaugos apibendrintas modelis, pabrėžiant svarbų konceptų, įvardytų terminais „paslaugos sąsaja“, „paslaugos realizacija“, „kontraktas“. Parodoma, kad dažnai į paslaugos modelį yra įtraukiami ir paslaugų sistemos architektūros elementai. Šio darbo tikslas – apibrėžti apibendrintą paslaugos modelį ir parodyti, kad jis neturėtų apimti architektūrinius sprendimus lemiančių elementų, nes tai perpina sistemų gyvavimo ciklo skirtingų stadijų turinius.

Pagrindiniai žodžiai: *programinė paslauga, paslaugos modelis, paslaugų architektūros stiliaus sistema.*

Įvadas

Kompiuterinės sistemos tampa ne tik vis sudėtingesnės, bet ir keičiasi jų ypatumai – siekiama sukurti nesunkiai modifikuojamas, patikimas, robusiškas sistemas, mažinti jų kūrimo trukmę. Iš paslaugų sudarytos sistemos yra naujo pobūdžio sistemos, kurių pagrindinis konstrukcinis elementas – stambaus grūdėtumo paslaugos, teikiamos bet kokiam naudotojų skaičiui. Be to, naujas sistemų kūrimo modelis charakterizuojamas „visur esančia“ sąveika su paslaugomis, todėl didesnę prasmę turi loginės, o

ne techninės ar fizinės kuriamos sistemos ribos.

Terminas „paslauga“ vartojamas įvairiuose kontekstuose ir sprendžiant skirtingas kompiuterinių sistemų kūrimo problemas. Daug painiavos kelia straipsniuose pateikiamos įvairios paslaugų apibrėžtys ir skiriamos paslaugų rūšys. Tokia įvairovė neišvengiama bet kurios paradigmos pradinėje formavimosi stadijoje, tačiau svarbu suprasti, kad reikia išsiaiškinti naudojamų paslaugos modelių skirtumus, nes jie lemia tiek paslaugų kūrimo, tiek paslaugų architektūros stiliaus sistemų (angl. *Servi-*

ce-Oriented Architecture) ypatumus. Standartizavimo problemas sprendžiančios organizacijos, tokios kaip OMG, OASIS, W3C, yra sudariusios paslaugų architektūros stiliaus sistemų pavyzdinių modelių ir ontologiją, paslaugos metamodelį, apibrėžė skaitymo paslaugų architektūrą, ir jų įnašas neabejotinas. Tačiau deklaruoti vieno supratimo, bendros kalbos sukūrimo tikslai vis dar yra tik iš dalies įgyvendinti. Šie klausimai yra nuolat aptariami moksliniuose straipsniuose (Portier, 2007; Cace, 2008; Lewis, 2010) ir diskusijų forumuose.

Šiame darbe nagrinėjami paslaugų modeliai – analizuojamos paslaugų apibrėžtys, aptariamas ir lyginamas panašių elementų, priklausančių skirtingiems modeliams, turinys. Naudojama paslaugos samprata turi būti aiški dėl dviejų svarbiausių priežasčių. Pirma, paslaugos ir jų specifikacijos turi būti vienareikšmiškai suprantamos kompiuterizuotos sistemos kūrėjų, nes jie turi sudaryti paslaugų rinkinį arba pasinaudoti jau egzistuojančiu paslaugų rinkiniu kaip pagrindu kompiuterinei sistemai sukurti. Antra, kad galėtų gauti ne tik paslaugą, bet ir tą rezultatą, kurio nori, naudotojas turi aiškiai suprasti ir pasirinkti tinkamą paslaugą. Pirmoje šio straipsnio dalyje pirmiausia aprašoma naudojama paslaugų modelių analizės metodika. Vėliau pateikiamos paslaugų apibrėžtys ir aptariami plačiausiai žinomi paslaugų modeliai. Nusakomas programinės paslaugos apibendrintas modelis, pabrėžiant terminais „paslaugos sąsaja“, „paslaugos realizacija“, „kontraktas“ įvardytų konceptų svarbą. Antroje straipsnio dalyje aptariami su paslaugų architektūros stiliumi susiję konceptai, parodoma jų ir projektinių sprendimų priklausomybė. Pabaigoje pateikiamos išvados.

Šio darbo tikslas – apibrėžti programinę paslaugą sudarant apibendrintą jos modelį ir parodyti, kad paslaugos modelis neturėtų apimti architektūrinius sprendimus lemiančių elementų, nes tai perpina sistemų gyvavimo ciklo skirtingų stadijų turinius.

Programinių paslaugų modelių analizė

Paslaugos yra esminė kompiuterizuotų paslaugų sistemų konstrukcija. Priežasčių, dėl ko literatūroje randama daug paslaugos sampratų ir apibrėžčių, yra kelios: a) paslaugos naudojamos skirtingų rūšių sistemoms kurti – programų, informaciniams ar techniniams; b) net ir vienos sistemos kontekste skiriamos skirtingų rūšių, kurios aptartos Besson, Čaplinsko (2009), Lupeikienės, Bagušytės (2009) darbuose, paslaugos; c) paslaugos nagrinėjamos kaip skirtingų sistemos gyvavimo ciklo stadijų artefaktai. Šiame skyriuje aptarsime paslaugos sampratą ir naudojamus paslaugų modelius.

Paslaugų modelių analizės metodika

Analizuojant literatūroje aprašytus paslaugų modelius taikomas koncepcinės analizės metodas. Jis leidžia nustatyti konceptų prasmę, apibrėžiant sąlygas, kurias turi tenkinti esybė, kad galėtų būti laikoma nagrinėjamu konceptu. Kitaip sakant, nustačius definityvines konceptų savybes, galima palyginti skirtingus literatūroje aprašomus rezultatus. Ryšiams tarp konceptų ir jų prasmei nustatyti taikomas ryšių analizės (angl. *relational analysis*) metodas. Nagrinėjamų konceptų ir ryšių panašumų bei skirtumų nesiekama įvertinti kiekybiškai, tiksliai pamatuojant jų artumą. Toks paslaugų modelių ir architektūrinių spren-

dimų tyrimas dar gali būti apibūdinamas kaip kokybinis ir stebimasis.

Nors nagrinėjama tyrimų sritis santykinai nauja, literatūros yra labai daug. Todėl siaurinant analizuojamų šaltinių paieškos sritį daroma prielaida, kad reprezentatyvią paslaugų modelių aibę galima sudaryti į ją jungiant paslaugų modelius, kuriuos sudarė a) standartizavimo problemas sprendžiančios organizacijos – OMG, OASIS, W3C, The Open Group, b) įmonės, vykdančios tyrimus ir kuriančios programinius produktus bei įrankius paslaugų sistemoms, pavyzdžiui, „Microsoft“.

Paslaugų modeliai ir jų palyginimas

Paslauga yra esminis paslaugų sistemų elementas. Paslaugų modelių nagrinėjimas šiame darbe iš pirmo žvilgsnio gali pasirodyti perteklinis ir nereikalingas. Tačiau kuriantieji paslaugų sistemas ir rašantieji apie jas vartoja skirtingas kalbas, tais pačiais terminais įvardija šiek tiek skirtingus turinius, turi skirtingus požiūrius. Tai ne tik sukelia neaiškumų, bet ir klaidina.

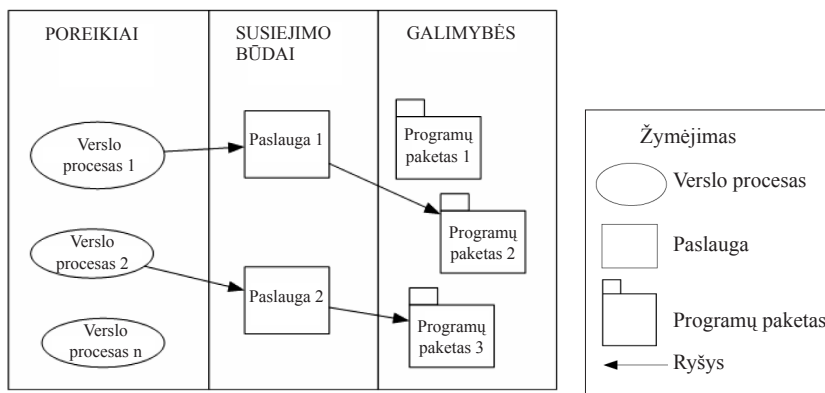
Paslauga vadinsime būdus, kuriais siejami poreikiai ir galimybės (1 pav.), o programinę paslaugą apibrėšime pateikda-

mi apibendrintą paslaugos modelį, kuriam nusakyti parinkta minimali nesusiklojančių konceptų aibė.

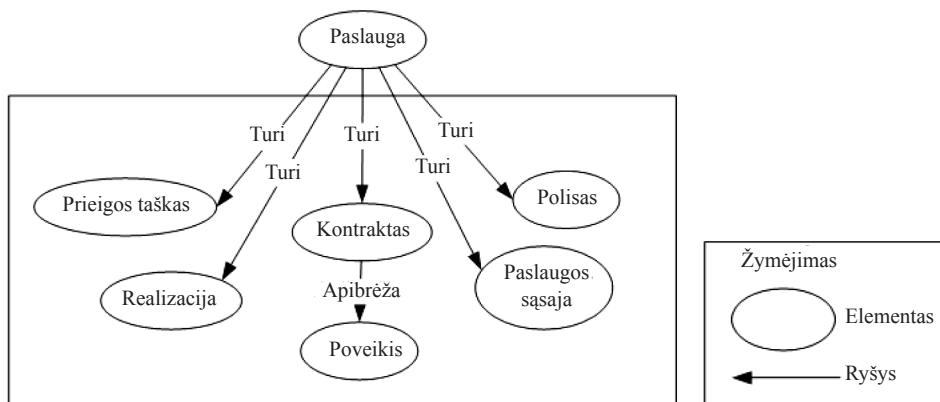
Paslaugos yra skirstomos į verslo ir viešąsias paslaugas, į skirtas naudotojams ir naudojimosi turimais ištekliais (infrastruktūros) paslaugas, į bazines, tarpininkavimo, procesų (angl. *process-centric*) ir išorei teikiamas paslaugas. Tačiau nepriklusomai nuo to, kokio tipo yra paslauga, ji visada sudaryta iš sąsajos ir realizacijos. Kad būtų galima pasinaudoti paslauga, turi būti sudarytas kontraktas ir žinomas prieigos (angl. *port, end point*) taškas. Apibendrintam paslaugos modeliui paaiškinti pateiksime konceptų apibrėžtis, kurios atspindi ir pasirinktą koncepcinį požiūrį.

Pritaikę koncepcinės analizės ir ryšių analizės metodus tiriamiems paslaugos modeliams, nusakysime programinės paslaugos apibendrintą modelį, pabrėždami konceptų įvardytų terminais „paslaugos sąsaja“, „paslaugos realizacija“ ir „kontraktas“, svarbą.

Siekdami vienareikšmiškai identifikuoti esminius nagrinėjamų paslaugų modelių skirtumus, toliau apibrėšime vartojamų konceptų reikšmes:



1 pav. Paslaugos samprata



2 pav. Programinės paslaugos apibendrintas modelis

Paslaugos interfeisas yra sąveikos su paslauga formos ir turinio specifikacija. Interfeise aprašomos teikiamos galimybės, konkrečiau – galimybių, nusakomų operacijomis, specifikacija ir pranešimų mainų protokolas, nurodantis, kaip pasinaudoti teikiamomis galimybėmis. Paslauga gali turėti kelis skirtingus interfeisus, kad ją būtų galima teikti pagal skirtingus protokolus.

Paslaugos realizacija vadiname paslaugą realizuojančią programą, kitaip tariant, programinį agentą, kuris realizuoja paslaugos sąsaja specifišką funkcionalumą, užtikrina reikiamas paslaugos savybes ir padaro numatytą poveikį pasauliui. Paslauga gali būti realizuota įvairiais būdais, skirtingomis programavimo kalbomis, tačiau bet kuri iš tų realizacijų turi užtikrinti visas sąsajoje aprašytas paslaugos savybes.

Kontraktas yra susitarimas, kuriame apibrėžtos paslaugos teikėjo ir gavėjo sąveikos taisyklės. Kontraktas aprašo paslaugos funkcionalumą ir jos kokybines charakteristikas, gavėjui keliamus reikalavimus, pranešimų struktūrą. Kontrakte

paprastai būna apibrėžtas ir paslaugos poveikis pasauliui.

Prieigos taškas yra vieta (adresas), į kurią kreipiamasi, kad būtų suteikta paslauga.

Programine paslauga vadiname kompiuteriniame tinkle teikiamą paslaugą, realizuojamą tam tikra programa. Sąveikos su paslauga forma ir turinys, atliekamos operacijos ir jų sekos, pranešimų mainų protokolas ir sąveikos taškai yra nurodomi paslaugos sąsajoje. Paslaugos teikėjas ir gavėjas sutaria dėl paslaugos teikimo / gavimo sąlygų ir numatomo rezultato, vadinamo poveikiu pasauliui, ir funkcionalumo, specifišką kontrakte. Paslaugos teikėjo ir gavėjo galimus veiksmus paslaugos teikimo / gavimo metu dar riboja politas, kuris papildo kontrakto sąlygas ribojimais, bendru atveju privalomais visiems kontrakto dalyviams.

Polisu vadinsime ribojimus, reglamentuojančius paslaugos dalyvių elgseną. Politą subjektyviai nustato paslaugos teikėjas siekdamas apriboti paslaugos gavėjo galimus veiksmus, ir atvirkščiai. Išskirtinė polito savybė, kad jo ribojimai gali būti

keičiami paslaugos teikimo metu. Jis nuskaidina tokias dinamines paslaugos savybes kaip saugumas (pvz., autentifikacija, ryšio kriptavimas), auditas ir pan.

Paslaugos poveikiu pasauliui vadiname pamatuojamą paslaugos suteikimo rezultata.

Apibendrinant koncepcinės analizės metodo taikymo pasirinktiems paslaugų modeliams rezultatus 1 lentelėje pateikiami paslaugų modeliai ir jų palyginimas: ištirtų modelių pagrindinių konceptų („paslaugos sąsaja“, „paslaugos realizacija“, „kontraktas“ ir „prieigos taškas“) reikšmės ir pabrėžti jų skirtumai kitų tiriamuose paslaugos modeliuose vartojamų konceptų („paslaugos teikėjas“, „paslaugos gavėjas“, „polisas“, „poveikis pasauliui“) kontekste.

Nagrinėjamos programinių paslaugų sampratos iš esmės nesiskiria, tačiau skirtingose apibrėžtyse akcentuojami skirtingi jų ypatumai.

OASIS pateiktoje pavyzdinėje paslaugų stiliaus sistemų architektūroje (Estefan ir kt., 2009) nusakant paslaugą pabrėžiamas poreikių ir galimybių susiejimo mechanizmas, t. y. paslauga – tai galimybių ir poreikių susiejimo būdai.

„The Open Group“ sudarytoje paslaugų architektūros stiliaus ontologijoje (OpenGroup, 2010) pabrėžiami du aspektai – paslauga yra veikla, teikianti specifikuotą rezultatą, ir paslauga yra loginė veiklos išraiška. Apibrėždami paslaugą „The Open Group“ akcentuoja paslaugos naudojamumą, paslaugos komponuojamumą ir paslaugos atstovavimą.

OMG, pristatydami paslaugų architektūros modeliavimo kalbą SoaML, pateikė paslaugos modelį (Berre, 2008), kuriame pabrėžiama paslaugos teikimo galimybė.

Kitaip tariant, paslauga matoma kaip gebėjimai ją teikti.

W3C parengė pasaulinio saityno paslaugų architektūrą (Booth ir kt., 2004) apibrėždami jos funkcinius elementus ir jų ryšius. Nors nuskaidydami paslaugą W3C pabrėžia, kad paslaugą jie mato kaip abstraktų išteklių, labai pabrėžiamas ir paslaugos užduoties vykdymo aspektas. Paslauga, kaip ir OMG, išreiškiama pajėgumu atlikti užduotį.

Palyginus tiriamų modelių konceptų apibrėžimus galima pastebėti nedidelių skirtumų, kurie daro įtaką tolesniems projektavimo sprendimams. Prieigos taškai vienuose paslaugos modeliuose aprašyti paslaugos apraše, kiti modeliai siūlo prieigos taškus aprašyti paslaugos sąsajoje. Polisas vienu atveju apibrėžiamas kaip papildomas veiksmų ribojimas, galiojantis bendram paslaugos gavimo / teikimo procesui, o kitu atveju jis laikomas subjektyviomis paslaugos gavėjo ir paslaugos teikėjo veiksmus ribojančiomis sąlygomis, kurios prieš teikiant paslaugą yra suderinamos ir įtraukiamos į kontrakto sąlygas, t. y. polisas laikomas kontrakto dalimi. Įdomus „The Open Group“ pasiūlymas užuot projektavus paslaugos teikėjui ir paslaugos gavėjui atskiras esybes, kaip kituose modeliuose, naudoti paslaugos dalyvio esybę, o kuris iš dalyvių atlieka kokį vaidmenį, nurodoma tik pasirašant kontraktą, t. y. sistemos realizavimo etape, pradėjus teikti programinę paslaugą. Paslaugos poveikis pasauliui vienuose iš nagrinėtų paslaugų modelių apibrėžiamas kontrakto sąlygose, kituose jis nuskaidomas bendrame paslaugos apraše. Vienu atveju poveikis pasauliui yra garantuojamas, o kitu atveju patogiau projektuoti paslaugos tarpininką su automatišku paslaugos pasirinkimu.

Lentelė. Paslaugų modeliai ir jų konceptų palyginimas

	Konceptas	Aprašymas
W3C	Paslaugos sąsaja	Sąsaja apibrėžia pranešimų tipus ir apsikeitimo jais eigą.
	Paslaugos aprašas	Tai dokumentų, aprašančių paslaugos sąsają ir semantiką rinkinys.
	Kontraktas	Kontraktas išreiškia bendrą paslaugos teikėjo ir paslaugos gavėjo susitarimą dėl jiems atstovaujančių programinių agentų sąveikos. Čia apibrėžiamas poveikis pasauliui.
	Prieigos taškas	Specifinė vieta pasiekti paslaugą naudojant specifinį protokolą ir duomenų formatą.
OASIS	Paslaugos sąsaja	Tai priemonė, sąveikauti su paslauga. Ji apibrėžia pranešimų semantiką, struktūrą. Čia išvardyti galimi paslaugos dalyvių veiksmai ir operacijos. Specifikuojami galimi įvykiai, paslaugos orkestravimo, choreografijos savybės.
	Paslaugos aprašas	Paslaugos aprašas yra dokumentas ar dokumentų rinkinys, apibrėžiantis informaciją, reikalingą paslaugos gavimui, skelbimui, naudojimui ir valdymui. Jį sudaro ne tik sąsaja, bet ir kita informacija, padedanti nuspręsti dėl paslaugos tinkamumo, – protokolas, sąveikos taškas, paslaugos funkcionalumas, politas, kontraktas.
	Kontraktas	Paslaugos dalyvių susitarimas. Prieš kontrakto pasirašymą prie kontrakto sąlygų yra derinami dalyvių politai.
	Prieigos taškas	Vieta, aprašyta paslaugos apraše, kur turi būti adresuojami pranešimai sąveikaut su paslauga. Kiekvienam paslaugos atliekamam veiksmui yra savas prieigos taškas ir su juo susietas protokolas.
Open Group	Paslaugos sąsaja	Dalyvių sąveikos ir keitimosi informacija su paslauga būdas.
	Dalyvis	Paslaugos dalyvis gali būti teikiantis paslaugą arba ją gaunantis. Jis gali teikti ar gauti keletą paslaugų. Jei paslaugą teikia – jis paslaugos teikėjas, jei gauna – paslaugos gavėjas.
	Atstovas	Vieta (adresas), kur kreipiamasi užsakant paslaugą (vieną ar kelias pasirinktinai).
	Kontraktas	Susitarimas tarp dviejų ar daugiau šalių. Kontraktas nustato, kuris iš dalyvių yra paslaugos gavėjas, kuris paslaugos teikėjas. Kontrakte specifikuojamas poveikis pasauliui.
OMG	Paslaugos sąsaja	Sąsaja nusako sąveikos ir komunikavimo protokolą, kaip tinkamai naudoti ir realizuoti paslaugą. Čia apibrėžiama dalyvių atsakomybė teikti ir naudoti paslaugą. Čia specifikuojami ir prieigos taškai.
	Kontraktas	Yra paslaugos teikėjo ir gavėjo susitarimo specifikacija, kokia informacija, produktais, verte ar įsipareigojimais bus keičiamasi. Čia specifikuojamas paslaugos funkcionalumas, sąveikos sąsajos, choreografija ir politai.
	Prieigos taškas	Prisijungimo vieta, kur konkreti paslauga yra užsakoma.
	Paslaugos taškas	Vieta (adresas), kur teikiama konkreti paslauga.

Visi šie skirtumai vienaip ar kitaip atsispindi sistemos architektūroje, jeigu ji kuriama pasirinkus konkretų paslaugos modelį. Šią įtaką architektūriniam sprendimams aprašysime kitoje straipsnio dalyje.

Palyginus tiriamus paslaugos modelius su pirmiau apibrėžtu apibendrintu programinės paslaugos modeliu (jis vaizduojamas 2 pav.), pagrindiniai tirtų modelių elementų turinių skirtumai pateikti lentelėje „Paslaugų modeliai ir jų palyginimas“.

Paslaugos modelių ir projektinių sprendimų priklausomybė

Detaliai išnagrinėję pristatytus paslaugų modelius galime nurodyti jų ypatybes ir galimą įtaką tolesniems architektūriniam sprendimams jų realizacijos etape. Nagrinėjamas architektūros sprendimo pasirinkimo aspektas schemiškai pavaizduotas 3 paveiksle.

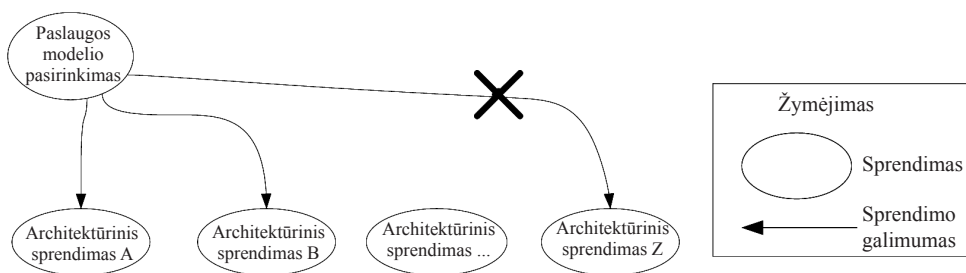
Dėl skirtingai suprantamų ir aprašomų paslaugos modelių konceptų skiriasi projektuojamų sistemų architektūra. Jei „The Open Group“ pavyzdžiu atsisakysime paslaugos gavėjo ir paslaugos teikėjo konceptų, turėsime projektuoti paslaugos dalyvio esybę. Su tuo susiję saugos klausimai, nes architektūrinis sprendimu iš anksto neapribojus paslaugos gavėjo veiksmų gauti paslaugą, tuos ribojimus

vėlesniuose projektavimo etapuose reikia realizuoti papildomais komponentais ar kitais programiniais sprendimais. Atsiranda dalyvio vaidmenys, jų priskyrimo ir valdymo komponentai. Sistemos architektūros lygmeniu keičiasi sistemos saugumo realizavimo metodai.

OASIS ir W3C išsiskiria tuo, kad modelyje naudoja paslaugos aprašą. Šis elementas lemia, kad atitinkamas dokumentas ar dokumentai su paslaugos aprašu atsirastų būsimoje sistemoje. Reikia numatyti, kaip tas dokumentas bus sudaromas (rankiniu būdu, automatiškai ar gal net bus pasinaudota specialia paslauga), kur jis bus įkeliamas ir saugomas, kas jį valdys. Visa tai yra sistemos projektavimo uždaviniai, išskylantys pasirinkus tam tikrą paslaugos modelį.

Kai paslaugos poveikis pasauliui yra apibrėžiamas ne kontrakte, o paslaugos apraše, projektuojant sistemą, automatiškai reikia į architektūrą įtraukti papildomus komponentus, pavyzdžiui, tarpininką – skelbimo lentą, kuris atliks veiksmus su paslaugos aprašu, pavyzdžiui, skelbs siūlomos paslaugos aprašą. Panaši situacija susidaro skirtingose vietose projektuojant paslaugos prieigos taškus.

Polisas, kurį iš anksto apibrėžia paslaugos teikėjas, paslaugų sistemos ar



3 pav. Architektūrinio sprendimo pasirinkimas

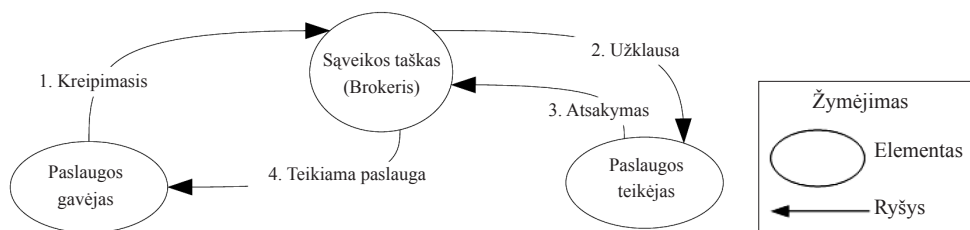
chitektūroje veikia kaip bendras, visiems architektūriniais sluoksniais galiojantis komponentas. Jei būtų vadovaujamasi paslaugos modeliu, kur polisai yra derinami, keičiami, pritaikomi ir suvedami į kontrakto sąlygas, tai sistemos architektūroje poliso komponentas dalyvautų kaip kontrakto komponento sudedamoji dalis ir paprastai veiktų tik paslaugos kokybės valdymo lygi.

Poliso reikalavimų integravimas į kontrakto sąlygas mažina paslaugų sistemos lankstumą, netenkama galimybės naudoti dinamines paslaugos savybes. Šis išskirtinis bruožas – keisti poliso reikalavimus paslaugos teikimo metu, suteikia paslaugų sistemoms pranašumą, palyginti su objektine ar komponentine paradigma. Šią galimybę apribojus, sistemos architektas turės atitinkamai pakeisti paslaugų sistemos architektūrą, įtraukti papildomus sisteminius komponentus ar kitais architektūriniais sprendimais kompensuoti dinamiškumo trūkumą.

Dar viena grupė paslaugos modelio sudarymo sprendimų, kurie daro įtaką architektūrai, yra į programinės paslaugos modelį įtraukiami elementai, kurie vėlesniuose projektavimo etapuose riboja sistemos architektūrinių sprendimų pasirinkimą.

„The Open Group“ akcentuojamas paslaugos atstovavimas tolesniuose įgyvendinimo žingsniuose turės įgyti papildomo elemento „Paslaugos atstovas“ pavidalą, ir šis elementas atliks papildomo paslaugos tarpininko ir sąveikos taško vaidmenį. Tokiu elemento pavyzdžiu galėtų būti „The Open Group“ paslaugos modelyje naudojamas sąveikos taškas (Atstovas), kuris sukuria papildomą prieigos iki paslaugos sluoksnį. Pagal šį modelį paslaugos gavėjas kreipiasi būtent į sąveikos tašką, o jau ten sprendžiama, kokią programinę paslaugą, kuri paslaugos funkcionalumą, kurios versijos ir per kurią sąsają paslauga bus suteikta. Tokiu būdu mes gauname papildomo lankstumo teikti paslaugų įvairovę ir siekti kokybės rodiklių, bet architektūriškai su paslaugos atstovo atsiradimu apribojame paslaugos teikimo schemas pasirinkimą. Į paslaugą kreipiamasi tik per paslaugos atstovą, o teikiama per ten pat, arba paslaugos gavėjas nukreipiamas į konkretų prieigos tašką, kur realiai teikiama paslauga. Šie sprendimai schemiškai pavaizduoti 4 paveiksle.

Šis elementas (Atstovas – Sąveikos taškas), atsiradęs paslaugos modelyje, riboja sistemos projektavimo galimybes, nes įtraukia papildomą paslaugos tarpininką.



4 pav. Paslaugos tarpininkas (Brokeris)

Be to, vadovaujantis „The Open Group“ paslaugos modeliu iš visų galimų atstovo sprendimų: Piršlio (angl. *Matchmaker*), Verbuotojo (angl. *Recruiter*), Brokerio (angl. *Broker*), Tarpininko (angl. *Facilitator*) ir kitų, sistemos architektūroje iš anksto numatoma įgyvendinti būtent brokerio sprendimą. Toks architektūrinių sprendimų ribojimas gali sukelti problemų įgyvendinant sistemai iškeltus reikalavimus.

Savaime suprantama, kad elementai „Paslaugos tarpininkas“ (angl. *Mediator*), „Paslaugos įgaliotinis“ (angl. *Proxy*), „Įmonės paslaugų magistralė“ (angl. *Enterprise Service Bus*), kuriuos labai mėgsta tie, kurie praktiškai kuria ir diegia programinių paslaugų sistemas į paslaugos modelį, neturi būti įtraukti, nes jie yra sistemos architektūros komponentai.

LITERATŪRA

BERRE, A. J. (2008). *Service oriented architecture Modeling Language (SoaML) – Specification for the UML Profile and Metamodel for Services (UPMS)* [interaktyvus]. Revised Submission OMG document: ad/2008-08-04 [žiūrėta 2010 m. rugsėjo 3 d.]. Prieiga per internetą: <<http://www.scribd.com/doc/9177449/SOAML-Specification-UML>>.

BESSION, J.; ČAPLINSKAS, A. (2009). Conceptual analysis of the notion of quality of services. *Lietuvos matematikos rinkinys*. Lietuvos matematikų draugijos darbai, t. 50, p. 357–361.

BOOTH, D.; HAAS, H.; McCABE, F.; NEWCOMER, E.; CHAMPION, M.; FERRIS, Ch.; ORCHARD, D., eds. (2004). *Web Services Architecture* [interaktyvus]. W3C Working Group Note 11 February [žiūrėta 2010 m. rugsėjo 3 d.]. Prieiga per internetą: <<http://www.w3.org/TR/ws-arch/>>.

CACE, B. (2008). SOA Terminology and the “SOA Reference Model” of OASIS. *Via Nova Architecture*, January, p. 1–7.

Išvados

1. Paslaugų stiliaus kompiuterinių sistemų teorija turi gana trumpą istoriją, todėl standartizavimo, pavyzdinių modelių, unifikuotų kalbų sukūrimo tikslai yra įgyvendinti tik iš dalies.
2. Apibendrintu programinės paslaugos modelių palygintų ir tirtų konkrečių paslaugos modelių pavyzdžiai patvirtina hipotezę, kad naudojami paslaugos modeliai gali lemti projektinių sprendimų aibę (pavyzdžiui, „The Open Group“ paslaugos modelis).
3. Straipsnyje buvo parodyta, kad paslaugos modelis neturi apimti architektūrinius sprendimus lemiančių elementų, nes sistemos projektiniai sprendimai privalo užtikrinti sistemai keliamus reikalavimus, o šios taisyklės nesilaikymas peršina sistemų gyvavimo ciklo skirtingų stadijų turinius.

ESTEFAN, J. A.; LASKEY, K.; McCABE, F. G.; THORNTON, D., eds. (2009). *Reference Architecture Foundation for Service Oriented Architecture* [interaktyvus]. Version 1.0, Committee Draft 02. OASIS, 14 October [žiūrėta 2010 m. rugsėjo 3 d.]. Prieiga per internetą: <<http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra-cd-02.pdf>>.

LEWIS, G. (2010). *Getting Started with Service-Oriented Architecture (SOA) Terminology* [interaktyvus]. Software Engineering Institute. Carnegie Mellon University, September [žiūrėta 2010 m. rugsėjo 3 d.]. Prieiga per internetą: <http://www.sei.cmu.edu/library/assets/whitepapers/SOA_Terminology.pdf>.

LUPEIKIENĖ, A.; BAGUŠYTĖ, L. (2009). Paslaugų klasifikavimas informacinių sistemų kūrimo kontekste. *Informacijos mokslai*, t. 50, p. 287–293.

OPEN GROUP (2010). *Service-oriented architecture ontology* [interaktyvus]. Technical Standard. The Open Group. 92 p. US ISBN 1931624887

[žiūrėta 2010 m. rugsėjo 3 d.]. Prieiga per internetą: <www.opengroup.org/bookstore/catalog/c104.htm>.

PORTIER, B. (2007). *SOA terminology overview*. [interaktyvus]. Part 1: Service, architecture,

governance, and business terms. IBM [žiūrėta 2010 m. rugsėjo 3 d.]. Prieiga per internetą: <<http://www-128.ibm.com/developerworks/library/ws-soa-term1/index.html>>.

SERVICE MODELS AND THEIR RELATION TO DESIGN DECISIONS

Marius Kaziliūnas

S u m m a r y

The service-oriented paradigm is still a recent development approach to computerized systems, so there are a wide variety of definitions of its concepts. In this paper, we focus on the service model. We analyse different service definitions and service model element definitions, which are discussed and compared with similar ele-

ments in different models. A software service model is presented, emphasizing the importance of the concepts identified by the terms “service interface”, “services realization”, and “contract”. The aim of the paper is to demonstrate that the service model should not include the elements that impact the further design decisions.