

ТЕХНОЛОГІЧНА МОДЕЛЬ ПРОЦЕСУ ПОБУДОВИ ТА ВИКОРИСТАННЯ АДАПТИВНОЇ КОМПОЗИЦІЇ WEB-СЕРВІСІВ

Запропоновано модель і технологічну схему процесу побудови й використання композиції Web-сервісів для різнорідних споживачів. Композитний сервіс подано динамічним сімейством його online-варіантів, описаних моделями: властивостей, порівневої варіабельності та ділових процесів і сервісів для елементарних властивостей. Процес побудови утворений операціями функцій управління варіабельністю (її планування, реалізації, моніторингу й актуалізації) цього сімейства в єдиному інформаційному середовищі, керованому моделлю варіабельності. Розроблені моделі разом уможливають залучення перспективних технік динамічного автоматизованого композування Web-сервісів як автоматичного планування через перевірку моделей, забезпечуючи порівневу контекстно-залежну адаптивність композитного сервісу, необхідну для його ефективного передбаченого використання.

Вступ

Однією з ключових переваг парадигми сервісно-орієнтованих обчислень [1, 2] є скорочення витрат і часу на розробку, розгортання і супровід складних розподілених застосунків, що являють собою (багаторівневі) композиції Web-сервісів, із збереженням контролю їх якості та перебігу життєвого циклу. Однак досягнення цієї переваги утруднюється через постійну змінність середовища виконання таких композицій: від наявності різнорідних сервісів, їх постачальників і споживачів з різними ролями, вимогами й перевагами до параметрів інфраструктури виконання. До того ж, переважна більшість змінних ситуацій невідома а ргіогі під час проектування композиції і може опрацьовуватися тільки під час її виконання.

Таким чином, критичним чинником якості й окупності композицій Web-сервісів стає їх адаптивність – “здатність до змін поведінки для задоволення нових вимог і пристосування до нових ситуацій” [3].

Аналіз стану справ у галузі побудови й використання адаптивних композицій Web-сервісів (АКС) на підставі доступних автору публікацій [2–11] висвітлює низку оригінальних підходів, де вагомих теоретичний доробок підтримано спеціалізованими інструментальними засобами:

1) редуцію побудови АКС до вирішення спеціальної проблеми автоматич-

ного планування через перевірку моделей (Центр Бруно Кеслера, Італія) [4–9];

2) розгляд АКС як динамічного сімейства його on-line варіантів (університет Монтеморелосу, Мексика) [10];

3) проектування АКС як віртуальної адаптивної організації (університет Свінбурну, Австралія) [11];

4) побудову й використання АКС на засадах повно-аспектного управління його варіабельністю (університет Свінбурну, Австралія) [2].

Однак аналіз наукового доробку підходів 1)–4) демонструє їх зосередження на окремих ситуаціях використання АКС:

– 1), 2) – “композування для себе”, тобто динамічне споживання всіх функцій самим компонувачем;

– 3), 4) – “композування для арендаторів”, тобто споживання окремих функцій АКС шляхом динамічного налаштування його on-line варіантів за моделлю SIMT [11].

Поза увагою залишається третя типова ситуація – ітеративне композування АКС і/або її on-line варіантів з іншими “зовнішніми” сервісами із створенням їх повнюваної мережі.

Узгоджена підтримка трьох зазначених ситуацій на всіх етапах життєвого циклу АКС складає мету роботи. Для її досягнення пропонується інтеграція підходів

1), 2), 4) і новітньої техніки аналізу впливів (Impact mapping) [12] для вилучення очікувань зацікавлених сторін АКС.

Стаття узагальнює результати автора в проєкті ДР 0112U002764 ІПС НАНУ.

Гібридний підхід до адаптивного композування Web-сервісів

Узагальнення результатів підходів 1)–4) дозволяє висунути вимоги до процесу побудови АКС, призначеної для (одночасного) використання у вищезазначених ситуаціях:

- уніфікація механізмів забезпечення адаптовності (придатності до змін архітектором під час проектування) та адаптивності (під час виконання);

- узгоджена реалізація контекстно-залежної адаптивності на всіх п'яти рівнях АКС [1] (потреб споживачів, ділових процесів, атомарних і композитних сервісів, реалізацій сервісів, операційних систем);

- забезпечення узгодженого реагування на зміни вимог і переваг користувачів з різними ролями, зміни статусу компонентних сервісів і контексту їх виконання;

- постійність і багаторазовість контекстно-залежних адаптаційних дій;

- врахування неповноти і обмеженої вірогідності поточної інформації про стан АКС і контексту.

Зіставлення висунутих вимог з рамковим життєвим циклом композування Web-сервісів [2] та рамковим каркасом моніторингу й адаптації Web-сервісу [3] висвітлює елементи каркасу (позначені на рис. 1), що мають підтримуватися в процесі побудови АКС, та пропозиції з їх реалізації. Останні подані у вигляді семи тез, що разом визначають сутність пропонованого підходу до побудови АКС.

T₁. Розгляд АКС як сімейства її online варіантів, спектр яких змінний при її виконанні (динамічного сімейства) [2, 10].

T₂. Подання АКС, як зазначеного сімейства, моделлю його варіабельності – здатності до ефективного розвитку, зміни, налаштування або конфігурування для використання в певному контексті [2, 10].

T₃. Використання в ролі початкового подання моделі варіабельності динамічної моделі властивостей (feature model, MB) [2] – дерева змінних та спільних властивостей (features) АКС із спеціальними відношеннями взаємозалежності, формально визначеними далі (див. означення 3).

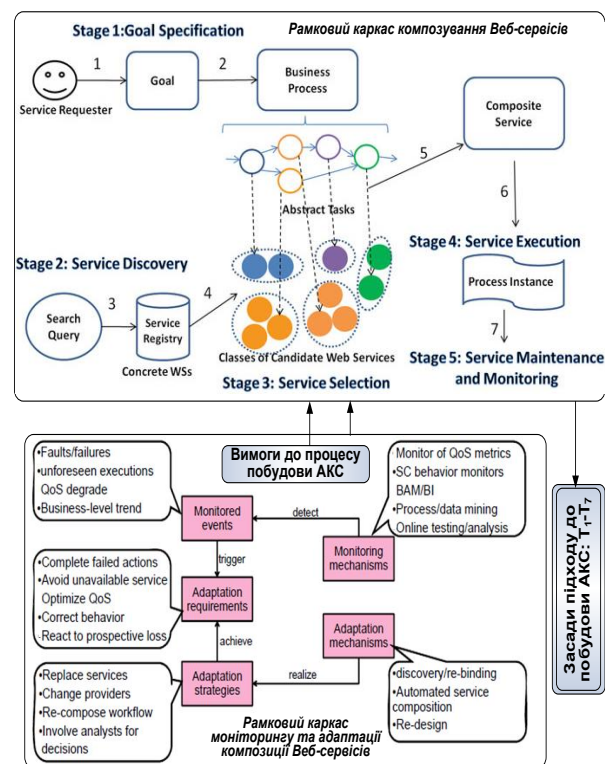


Рис. 1. Підстави підходу до побудови АКС

T₄. Інтеграція з підходом 1) до автоматизованого адаптивного композування шляхом зіставлення термінальним властивостям елементарних *доменних об'єктів* [5–8], призначених для відокремленого моделювання вимог до операцій і даних абстрактних ділових процесів на підтримку цих властивостей ($\rho_c \wedge \rho_d$) та до конкретних термінальних сервісів їх реалізації ($\rho^s_c \wedge \rho^s_d$).

T₅. Подання процесу побудови й використання АКС композицією спеціальних функцій управління варіабельністю в єдиному інформаційному середовищі, керованому моделлю властивостей, шляхом адаптації відповідної моделі для канонічного сімейства застосунків [13].

T₆. Редукція проблеми динамічного композування до проблеми автоматичного планування через перевірку моделей на

підставі поточної моделі властивостей та стану середовища виконання АКС.

Т7. Заміна подій – передумов адаптації, які в підході 1) мають визначитися архітектором АКС під час її проектування, апріорно подією незадовільності стану варіабельності АКС з подальшим формулюванням проблеми адаптації як проблеми динамічного перекомпозування через формальну перевірку моделей.

Формальний апарат адаптивного композування

Основні означення. Реалізація тез потребує взаємного налаштування підможин формального апарата інтегрованих підходів та уточнення моделі властивостей відповідно до п'ятирівневої структури АКС і особливостей варіабельності Web-сервісів. Сутність останньої фіксує

Означення 1 [2]. Варіабельність Web-сервісів – частковий випадок варіабельності програмного забезпечення для сервісно-орієнтованих програмних систем (СоПС). Це здатність Web-сервісу до ефективного розвитку, зміни, налаштування або конфігурування для використання в певному контексті.

Визначальними особливостями варіабельності Web-сервісів є:

а) три рольові погляди, властиві постачальнику й компоновнику сервісів та споживачу композитного сервісу;

б) динамічність і рекурсивність комунікацій щодо варіабельності СоПС;

в) три її взаємопов'язані типи:

– спостережувана (exposed), $k=1$;

– композиційна (composition), $k=2$;

– компонентна (partner), $k=3$;

г) п'ять рівнів забезпечення варіабельності, відповідні структурі СоПС [1].

Спостережувана варіабельність призначена для опису припустимих змін операцій, протоколів і типів повідомлень в інтерфейсі композитного сервісу. Для неї додатково виокремлюються три підтипи, які стосуються змін у складі запитуваних параметрів, в їх значеннях та в протоколах.

Композиційна варіабельність стосується логіки формування композитного

сервісу і відображає різні способи оркестрування компонентних сервісів для реалізації різних функціональних можливостей.

Нарешті, компонентна варіабельність описує змінність взаємодій між компонентними сервісами в складі композитного сервісу. Вона являє собою частковий випадок спостережуваної варіабельності компонентного сервісу (що сам може бути композицією композитних сервісів з довільним рівнем рекурсії).

Таким чином, спостережувана варіабельність:

– здійснює вплив на компонентну варіабельність, оскільки спостережуваний інтерфейс обумовлює вибір компонентного сервісу з його власною спостережуваною варіабельністю;

– істотно обумовлює композиційну варіабельність, яка залежить також і від компонентної варіабельності.

До числа рівнів забезпечення варіабельності СоПС належить:

1) система вимог, варіабельність яких подається MB чи розширеннями мови BPMN [2] ($l=1$);

2) визначення ділового процесу, описуване за допомогою варіабельно-орієнтованих розширень абстрактної мови BPEL або мови CVL і WSVL [2] ($l=2$);

3) опис інтерфейсу з використанням мови WSDL або її варіабельно-орієнтованого розширення WSVL [2] ($l=3$);

4) опис реалізацій сервісів і параметрів інфраструктури ($l=4, 5$).

Прояви варіабельності АКС типів $k=1, 2, 3$ на рівнях $l=1, \dots, 5$, “успадковані” від канонічної варіабельності, описує

Означення 2. Точка варіантності – формальне подання, в артефактах процесу розроблення СоПС, елемента ділового процесу, що може реалізуватися кількома способами.

Варіант (variant) для точки варіантності – спосіб реалізації елемента, який вона подає. Типи точок варіантності обумовлюють кількість (від 0 до всіх) варіантів, які можуть зв'язуватися з ними для певного варіанта Web-сервісу. Ефективним способом опису взаємозв'язку точки варі-

антності з її варіантами є пара $(0 \leq m \leq n)$ чисел кардинальності [2], які фіксують мінімальну й максимальну кількість зв'язаних варіантів.

Залежність (dependance) – відношення на множині точок варіантності й варіантів, яке обмежує вибір варіантів для деяких точок варіантності в залежності від їх вибору для інших точок.

Обмеження (constraint) – залежність, яка стосується тільки точок варіантності. Типовими обмеженнями є відношення потреби/виключення.

Отже, на відміну від статичної ситуації у канонічному сімействі програмних систем, АКС у ролі динамічного сімейства має особливості [2, 10], сприятливі для її динамічної адаптації:

а) моделі варіабельності можуть публікуватися та багаторазово застосовуватися агентами з різними ролями;

б) моделювання варіабельності по АКС потребує моделювання варіабельності її компонентних сервісів з довільним рівнем рекурсії;

в) можливі не тільки внутрішні (intra-) залежності між композованими сервісами, але й зовнішні (intra-) залежності між структурою композиції і компонентними сервісами, які самі є композитними, з довільним рівнем рекурсії.

Явне моделювання проявів варіабельності АКС типів $k=1,2,3$ на рівнях $l=1, \dots, 5$ для динамічного оцінювання рівня її задовільності згідно з T_7 потребує формального опису МВ за допомогою

Означення 3. МВ АКС

$$FM = (F, R_F), R_F = \{C_{mn}; IM; EQ; EX\} \quad (1)$$

– дерево її змінних та постійних властивостей $f \in F$, пов'язаних відношеннями, що утворюють множину R_F . Коренем МВ є задоволеність потенційних споживачів функціями АКС. Властивість f – змістовне формулювання, що подає потребу або очікування певної групи потенційних споживачів АКС.

Властивості $f_i, f_j \in F$ пов'язані:

1) (m,n) -варіантним підпорядкуванням $(C_{mn}, 0 \leq m \leq n \leq (|F|-1))$, якщо ная-

вність в АКС підпорядковуючої функції f_i впливає з реалізації всіх f_j , відповідно, від m до n з $VF \subset F$ підпорядкованих функцій f_j ;

2) обумовленістю IM чи зіставленням EQ , якщо реалізація f_i потребує f_j або, відповідно, вони можуть реалізуватися тільки разом;

3) виключенням EX , якщо реалізація f_i виключає реалізацію f_j .

При цьому IM, EQ, EX неприпустимі в групі VF властивостей, разом безпосередньо підпорядкованих одній властивості.

Подальшу конкретизацію МВ (1) із врахуванням особливостей АКС надає

Означення 4. Порівнева модель варіабельності АКС, відповідна її МВ FM (1), – структурована трійка

$$VM = \langle SV; AV; EV \rangle, \quad (2)$$

де SV і AV – порівневні моделі варіабельності в структурі та в артефактах АКС;

EV – діагностична модель рівня задовільності варіабельності АКС.

Підмоделі VM (2) надають розвиток для АКС своїх однойменних попередників [13], запропонованих для канонічного сімейства за участі автора. Сутність розвитку висвітлюють означення 5–7.

Означення 5.

Модель варіабельності в структурі АКС – тригілковий п'ятирівневий кортеж

$$SV = \langle \langle G_{1k}; \langle \langle G_{lk}, TR_{lk} \rangle, l=2, \dots, 5 \rangle \rangle; \quad (3)$$

$$CN_k; DP_k \rangle, k=1,2,3; CN; DP \rangle,$$

де $G_{lk} = (F_{lk}, R_{F_{lk}})$ – граф, вершинами якого є унікальні ідентифікатори проявів варіабельності типу k на рівні l , а дугами – відношення їх підпорядкування, зумовлені FM ;

TR_{lk} – двосторонні зв'язки трасованості між вершинами $G_{(l-1)k}$ і G_{lk} ;

CN_k, DP_k – предикати на $\otimes_{l=1, \dots, 5} F_{lk}$, що подають обмеження й залежності для варіабельності типу k ;

CN, DP – предикати, що подають обмеження й залежності між проявами варіабельності різних типів, визначені на $\otimes_{l=1,\dots,5, k=1,2,3} F_{lk}$.

Модель варіабельності в артефактах АКС надає уніфіковане формальне подання для всіх артефактів динамічного сімейства, припустимих за його поточного стану. Цей артефакт розглядається як прояв варіабельності типу k рівня m і подається вертикальним наскрізним фрагментом SV (3).

Означення 6. Подання прояву варіабельності АКС типу k рівня m з ідентифікатором id_{mk} – структурований кортеж

$$AV(id_{mk}) = \langle g_{lk}; \langle \langle g_{lk}, tr_{lk} \rangle; l=2, \dots, m \rangle; \langle \langle p_{lk}, tr_{lk} \rangle; l=m+1, \dots, 5 \rangle; cn_k; dp_k \rangle, \quad (4)$$

$$k = 1, 2, 3; cn; dp \rangle,$$

де g_{uk} і p_{uk} – підграфи G_{lk} з (3), а решта елементів – відповідні звуження елементів SV (3).

Означення 7. Діагностична модель ступеню задовільності варіабельності АКС – пара

$$VLM = (EV, CA), \quad (5)$$

$$EV = \langle il_k; \langle vl_{lk}, rl_{lk}, pl_{lk}, ml_{lk} \rangle; ia_k; \quad (6)$$

$$\langle va_{lk}, ra_{lk}, pa_{lk}, pu_{lk} \rangle; \langle NA_{lik}, l, i=1, \dots, 5 \rangle; k=1, 2, 3 \rangle,$$

де EV – рівень задовільності варіабельності АКС для всіх зацікавлених сторін;

il_k, ia_k – інтегральні рівні автономної варіабельності АКС та її відповідності потребам його споживачів і постачальників з різними ролями;

vl_{lk} і va_{lk} – проміжні рівні автономної варіабельності типу k та її відповідності, формовані артефактами l -го рівня АКС;

rl_{lk}, pl_{lk} і ml_{lk} – вкладені рівні варіабельності типу k , передбаченої й реалізованої для артефактів l -го рівня АКС, а також ступеню відповідності між ними;

$ra_{lk}, pa_{lk}, pu_{lk}$ – вкладені ступені адекватності передбаченої й реалізованої варіабельності типу k в артефактах l -го рівня АКС потребам її користувачів і роз-

робників, а також ефективності повторного використання артефактів;

NA_{lik} – множина артефактів l -го рівня АКС, що обумовлюють виявлену неадекватність типу i для варіабельності типу k ;

CA – перелік операцій над відповідними елементами NA_{lik} для усунення виявленої неадекватності.

Аналіз підходів до управління варіабельністю Web-сервісів [2, 10] висвітлює п'ять базових типів неадекватності:

1) “надлишковість” – наявність артефактів, незастосованих/неокупних під час побудови й використання АКС;

2) “неповнота” – відсутність в АКС функцій, запитаних значущою більшістю її користувачів, або компонентних сервісів для них;

3) “клони” – наявність артефактів, повністю взаємозамінних під час побудови й використання АКС;

4) “хибні обмеження” – наявність артефактів, яким потрібна зміна статусу як точки варіабельності;

5) “хибні залежності” – артефакти, що потребують перепідпорядкування і/або зміни типу як варіанти певної точки варіабельності.

Листки, вузли та інтегральні рівні моделі EV (5, 6) доповнюються експертно формованими вербально-числовими шкалами – наборами пар “опис стану АКС; рівень критичності”.

Ядро множини CA , відкрите для поповнення, складають три групи операцій:

1) автономна еволюція або заміна елементарних компонентних сервісів, відповідних термінальним властивостям FM (1), без зміни їх використовуваних описів абстрактною мовою WS-BPEL і WSDL;

2) долучення/вилучення функцій АКС та актуалізація їх співвідношень у FM, SV, AV без зміни листків FM . Ці операції здійснюються шляхом узгодженої актуалізації FM разом з графами G_{lk} в SV ;

3) узгоджена еволюція елементарних компонентних сервісів із змінами їх використовуваних описів та актуалізація

графів G_{lk} і листків FM включно з зіставленими їм доменними об'єктами.

Модель VM (2) дозволяє зіставити МВ, як подання АКС у просторі проблеми, її подання у просторі рішень. Це сукупність доменних об'єктів ($DO(t)$), що відокремлено моделюють вимоги до операцій [7] і даних [8] абстрактних ділових процесів на підтримку елементарних властивостей та подібні вимоги до відповідних термінальних сервісів. Таке розмежування додатково сприяє контекстно-залежній покроковій динамічній адаптації АКС.

Означення 8. Динамічна модель АКС, яка має МВ FM (1), у довільний момент часу t – кортеж

$$CS(t) = \langle FM(t); VM(t); DO(t); S(t); vm, va \rangle, \quad (7)$$

де $S(t)$ – віртуальний репозиторій термінальних сервісів, які реалізують ділові процеси на підтримку термінальних властивостей;

vm і va – постійно доступні сервіси моніторингу ступеню задовільності варіабельності АКС та її автоматизованої динамічної адаптації.

Постійними функціями vm є:

а) експрес-оцінювання ступеню задовільності варіабельності АКС за діагностичною моделлю EV (5),(6) і регламентом, заданим під час проектування;

б) діагностування незадовільності цього ступеня та виявлення її джерел NA_{lik} ;

в) вироблення коригуючих дій з усунення виявленої незадовільності й надання необхідної інформації адаптеру va .

Адаптер має автоматизовано виконувати вироблені дії щодо елементів моделі $CS(t)$ (7) з необхідним залученням архітектора АКС.

Первинна побудова моделі АКС (7) під час її проектування та її постійна актуалізація утворюють пропонований процес побудови й використання АКС, який розвиває процес управління варіабельністю традиційного сімейства [13].

Означення 9. Технологічна модель процесу побудови й використання АКС з

моделлю $CS(t)$ (7) – тригілковий структурований кортеж

$$EM(t) = \langle \langle RL, FN, EN(t) \rangle; TD; PS(t), MT(t); AT \rangle;$$

$$FN = \{F_i, i = 1, \dots, 4\}; F_i = \langle O_i, G_i \rangle,$$

$$G_i \subseteq O_i \otimes O_i; \quad (9)$$

$$\forall o \in O_i \quad o = \langle pp, a, in, ot, c, ps, m, t \rangle, \quad (10)$$

$$a \subseteq RL, in, ot, c \subseteq EN; ps \subseteq PS;$$

$$m \subseteq MT(t), s \subseteq AT,$$

де RL – ролі учасників процесу (постачальник, компонувач, споживач, орендар (tenant), брокер);

F_i – функції управління варіабельністю АКС, запроваджені в [13] для традиційного сімейства на підставі відомого циклу управління Е. Дьомінга;

$$EN = \langle \langle SV, AV \rangle; \langle FR, DR, RR \rangle; S; \langle EV, VP \rangle \quad (11)$$

моделльне середовище для функцій F_i ;

TD – раціональні вимоги до операцій цих функцій.

Функції управління варіабельністю включають:

1) планування реалізації варіабельності в структурі й артефактах АКС (F_1);

2) реалізацію запланованої (F_2);

3) системний моніторинг задовільності варіабельності АКС (F_3);

4) еволюцію АКС за результатами моніторингу (F_4).

Подання операції o (10) складене її призначенням (pp), ролями виконавців ($rl \subseteq RL$), входами, результатами й контекстом (in, ot, c) та формальними постановками ($ps \subseteq PS$), методами ($mt \subseteq MT$), розв'язання типових задач і засобами автоматизації ($s \subseteq AT$).

Склад і структура модельного середовища EN (11) у поточний момент t , визначається моделлю $SV(t)$ (3). Крім безпосередньо обумовлених нею моделей $AV(t)$ (4) і $EV(t)$ (5), (6), воно містить віртуальний репозиторій доступних термінальних сервісів $S(t)$, репозиторій профілів ступеню задовільності варіабельності

(*VP*) та репозиторії повторно використаних артефактів АКС: попередніх МВ (*FR*), доменних об'єктів (*DR*), вимог (*RR*).

Пропонована структура порівневої моделі варіабельності АКС (2) створює передумови підтримки раціональних вимог *TD* до операцій (10):

1) обґрунтованості – надання підстав прийняття рішень щодо функцій (*D*₁);

2) узгодженості способів вироблення й реалізації цих рішень на п'яти рівнях і на всіх етапах побудови АКС (*D*₂);

3) масштабності – незалежності способу вироблення й реалізації зазначених рішень від обсягу функціональних можливостей АКС (*D*₃);

4) трасовності – можливості простеження зв'язків між проявами варіабельності типів *k* на рівнях *l* і на всіх етапах процесу побудови АКС (*D*₄).

Формування варіабельних вимог до композиції на підставі моделі властивостей. Оскільки МВ (1) є за означенням деревом, її термінальні властивості (листки) незалежні між собою в тому сенсі, що жодна з них не може бути реалізована за допомогою решти термінальних властивостей. На відміну від них, властивості верхніх рівнів фактично являють собою поєднання обов'язкових і необов'язкових термінальних властивостей, додатково пов'язаних залежностями, що утворені відношеннями *IM*, *EQ*, *EX* різних рівнів моделі властивостей.

Таким чином,

$$F = FM \cup FO, FM \cap FO = \emptyset,$$

де $FM = \{f \in F \mid \exists! f^* C_{1,1}(f, f^*)\}$ – обов'язкові (mandatory) властивості, які мають реалізуватися для всіх споживачів за будь-яких умов використання;

$FO = \{f \in F \mid \exists! f^* C_{0,1}(f, f^*)\}$ – необов'язкові (optional) властивості, які реалізують потреби тільки окремих категорій споживачів.

Запровадимо до розгляду одномісний предикат *f* із сенсом "АКС має властивість *f*" (однаковість позначень збере-

жено для спрощення нотації). Відношення між властивостями в (1) проінтерпретуємо з точки зору логіки висловлювань:

IM – імплікація одномісних предикатів, відповідних його аргументам;

EQ – кон'юнкція таких імплікацій;

EX – імплікація між предикатом, відповідним першому аргументу, та запереченням другого.

Тоді МВ можна подати формулою висловлювання предикатів

$$r = RM \vee RO;$$

$$RO = \vee_{\{FP, \emptyset \subseteq FP \subseteq FO\}} (\wedge_{fo \in FP} fo \wedge Dep(FM, FP)); RM = (\wedge_{fm \in FM} fm) \wedge Dep(FM), \quad (12)$$

де $Dep(FM, FP)$ – кон'юнкція залежностей між обов'язковими і необов'язковими властивостями з *FP*, утворених відношеннями *IM*, *EQ*, *EX* у предикативній формі;

$Dep(FM)$ – кон'юнкція аналогічних залежностей обов'язкових властивостей.

Слід зазначити, що $Dep(FM, FP)$ і $Dep(FM)$ являють собою композиції елементарних предикатів *fo* і *fm*, утворені за допомогою логічних зв'язків $\vee, \wedge, \neg, \rightarrow$.

Користуючись відносною простою і автономністю термінальних властивостей, кожному елементарному предикату *fm* і *fo* зіставимо кон'юнкцію вимог до операцій і даних, пов'язаних з доменним об'єктом для цієї властивості:

$$\rho_m = \rho_{mc} \wedge \rho_{md} \wedge (\rho_{mc}^s \wedge \rho_{md}^s) = \rho_{mc}^* \wedge \rho_{md}^*; \quad (13)$$

$$\rho_o = \rho_{oc} \wedge \rho_{od} \wedge (\rho_{oc}^s \wedge \rho_{od}^s) = \rho_{oc}^* \wedge \rho_{od}^*.$$

Саме, вимоги до операцій (ρ_c^*) подамо за допомогою мови EaGLE [7], а вимоги до даних (ρ_d^*) – за допомогою абстрактної та анотаційної мереж даних (datanet) [8]. Слід зазначити, що внаслідок елементарного характеру функцій, поданих листками МВ, виконання умов ρ_m, ρ_o для кожної з них не залежить від решти.

Виконання операції заміни в предикативному поданні МВ (12) предикатів *fo* і *fm* їх поданнями ρ_o і ρ_m для автоматизованої композиції (позначимо її τ) дозволяє переписати його у вигляді

$$\rho = \tau(RM) \vee \tau(RO); \quad (14)$$

$$\tau(RM) = (\wedge_{fm \in FM} \rho_{mc} \wedge \rho_{md}) \wedge \tau(Dep(FM));$$

$$\tau(RO) = \vee_{\{FP, \emptyset \subseteq FP \subseteq FO\}} (\wedge_{fo \in FP} (\rho_{oc} \wedge \rho_{od}) \wedge \tau(Dep(FM, FP))). \quad (15)$$

Отримане подання (14), (15) надає необхідні варіабельні вимоги до АКС. У свою чергу, його елементарні складові ($\rho_{mc}^s \wedge \rho_{md}^s$) та ($\rho_{oc}^s \wedge \rho_{od}^s$) визначають, у необхідній для автоматизованого композування формі [7, 8], елементарні й незалежні між собою вимоги до конкретних компонентних сервісів. Простота й незалежність цих вимог сприятимуть полегшенню пошуку необхідних компонентних сервісів у реєстрах та безпосередньому створенню.

Технологічна схема побудови адаптивної композиції

Внутрішня структура схеми. На підтримку процесу побудови й використання АКС згідно з моделлю (9)–(11) пропонується трифазна технологічна схема, подана на рис. 2. Згідно з рисунком, схема являє собою технологічну конкретизацію композиції функцій управління варіабельністю АКС F_i , $i=1, \dots, 4$ в єдиному інформаційному середовищі EN (11).

Перша фаза схеми призначена для подання проблеми динамічної побудови АКС згідно з МВ (1), що описує поточні потреби його потенційних споживачів, як спеціальної проблеми автоматичного планування [7–9]. Вона включає вісім послідовних кроків.

Крок 1. Побудова поточної моделі властивостей FM (див. означення 3) як подання формованого АКС у просторі проблеми поточного динамічного сімейства його on-line варіантів.

Крок 2. Зіставлення термінальним властивостям (листкам) FM елементарних доменних об'єктів – технічних конструктів подання вимог до ділових процесів на підтримку властивостей та до елементарних компонентних сервісів їх реалізації.

Крок 3. Опис доменних об'єктів діаграмами станів [8].

Крок 4. Специфікація вимог зі складу (13) до операцій і даних на підтримку взаємодії доменних об'єктів.

Крок 5. Специфікація вимог щодо елементарних компонентних сервісів на підтримку взаємодії доменних об'єктів у вигляді анотованих описів де-факто стандартними мовами BPEL, WSDL $d_i = (b^*_i, w^*_i)$ на підставі (13).

Крок 6. Автоматизований пошук (у реєстрі UDDI, інших реєстрах, серед пропозицій постачальників) сервісів s_i , (b, w) -описи яких (безпосередні або формовані на підставі подань у UDDI [14]) не суперечать b^*_i, w^*_i , сформованим на Кроці 5.

Крок 7. Упорядкування отриманих множин $\{s_i\}$ за відповідністю описам d_i і нефункціональними характеристиками (QoS) згідно з сучасними методами [2];

Крок 8. Автоматизована верифікація перших елементів впорядкованих множин на відповідність вимогам до них, виробленим на Кроках 4, 5 за допомогою відкритих середовищ верифікації (Spin, NuSmv [15]).

Описи d_i верифікованих сервісів і узгоджена з ними відповідна їм (часткова) варіабельна ціль композиції ρ (14), (15) являють собою вхідні умови для другої фази.

Отже, перша фаза підтримує функцію планування варіабельності АКС (F_1) для забезпечення його адаптивності, опрацьовуючи виклики перших трьох етапів її життєвого циклу, зафіксованих на початку роботи (див. рис. 1).

Впродовж другої фази за допомогою вдосконалених технік перевірки моделей [7–9], автоматично формулюється й вирішується проблема автоматизованого композування сервісів з описами $\{d_i\}$ згідно з варіабельними вимогами ρ . Як показано на рис. 2, друга фаза охоплює чотири кроки:

1) побудова для компонентного сервісу з описом d_i системи з переходом між станами [9] СТС $S_i(d_i)$;

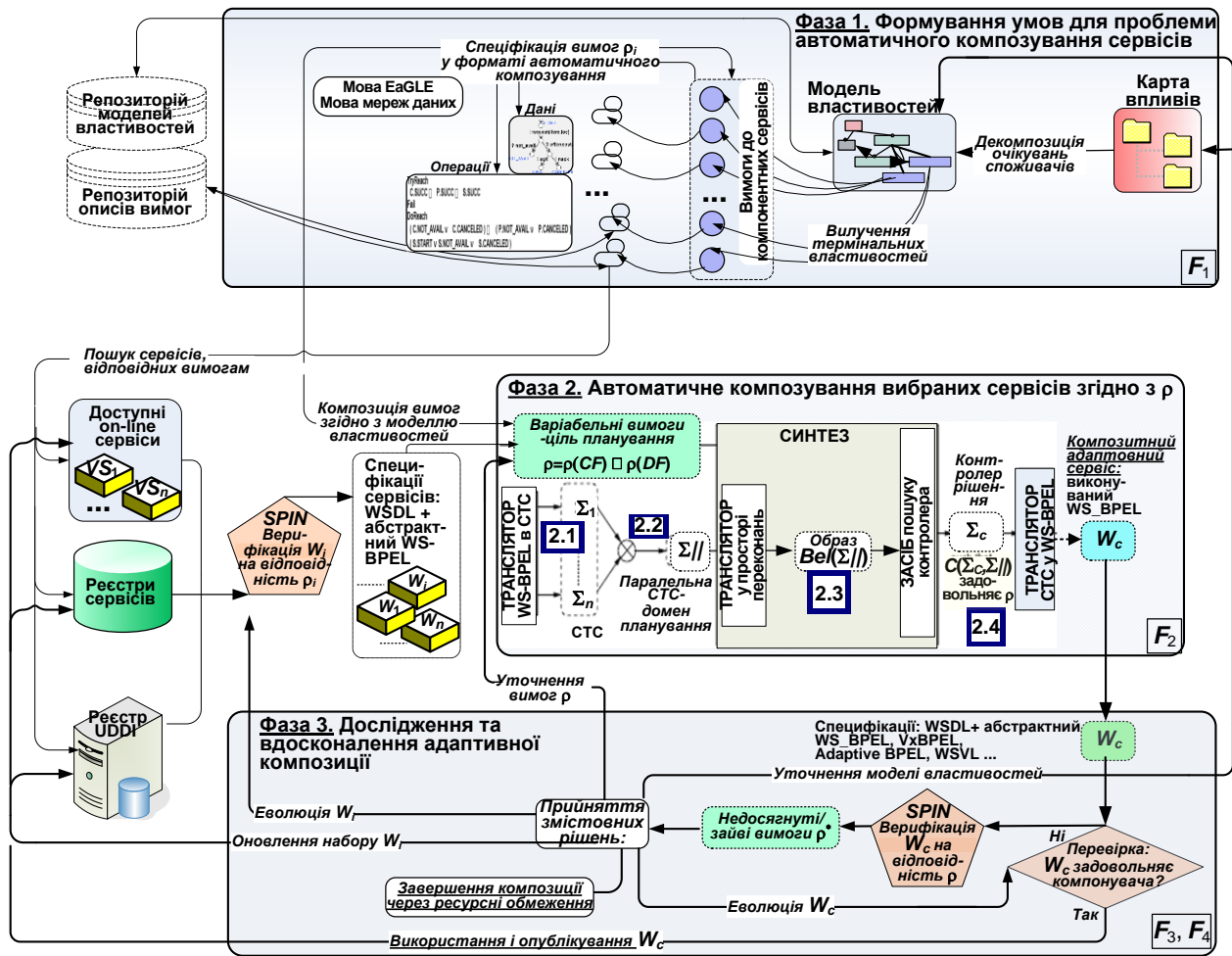


Рис. 2. Технологічна схема процесу побудови АКС

2) визначення СТС $S||$, яка є паралельним добутком СТС $S_i(d_i)$, та її образу на рівні переконань $Bel(S||)$;

3) дослідження структури виконання в СТС $Bel(S||)$ для ідентифікації одного піддерева виконання, асоційованого з контролером рішення проблеми.

4) вилучення контролера S_c з дерева виконання і зіставлення йому опису процесу (програми) виконуваною мовою WS-BPEL W_c .

Таким чином, підтримується функція реалізації варіабельності АКС (F_2) та опрацьовуються виклики композування Web-сервісів на четвертому етапі його життєвого циклу згідно з рис. 1.

Нарешті, на третій фазі постачальник здійснює дослідження поведінки і показників QoS та необхідне вдосконалення отриманого композитного сервісу W_c . За-

довільний сервіс надається передбаченим споживачам і може публікуватися в загальнодоступних реєстрах (UDDI тощо) через стандартизований опис $d(W_c)$, стаючи об'єктом адаптації часу виконання.

Для АКС, поведінка якої визнана незадовільною, оцінюється ступінь задовільності варіабельності за моделлю EV (5), (6) і перевіряється відповідність бажаним вимогам $r^*(r)$, уточненим на підставі NA_{ii} , за допомогою середовища SPIN. Із врахуванням виявлених недосягнутих вимог і NA_{ii} приймається підмножина типових рішень:

- 1) відмова від композиції через ресурсні обмеження;
- 2) еволюція W_c як атомарного сервісу згідно з підходом М. Папазоглу;
- 3) узгоджена еволюція моделі властивостей $FM(t)$ і набору компонентних

сервісів з описами d_i та повторення композивання.

Повторне використання адаптивного композитного сервісу. Запропоновані моделі АКС (7) та процесу її побудови й використання (9)–(11) полегшує:

1) адаптацію АКС під час виконання, тобто автоматичного вироблення й реалізації дій з дотримання вимог споживачів у разі їх порушення;

2) налаштування on-line варіантів композитного сервісу за моделлю його множинної оренди (SIMT) [11];

3) ітеративне композивання композитного сервісу і/або його on-line варіантів з іншими “зовнішніми” сервісами із створенням їх поповнюваної мережі.

Висновки

1. Обґрунтовано актуальність побудови адаптивної композиції Web-сервісів, орієнтованої на (одночасне) використання у трьох типових ситуаціях: споживання всіх її функцій самим компонувальником; налаштування окремих функцій орендарями; необмеженого ітеративного отримання нових функцій зовнішніми споживачами з формуванням мережі сервісів.

2. Запропоновано гібридні моделі адаптивної композиції Web-сервісів та процесу її побудови для цих ситуацій. Вони ґрунтуються на інтеграції підходів до розгляду композиції як динамічного сімейства її on-line варіантів та як результату динамічної контекстно залежної покрокової адаптації, редукованої до проблеми автоматичного планування для досягнення поточних цілей адаптації.

3. Розроблена модель композиції являє собою її подання, як динамічного сімейства, у просторах проблеми (моделлю властивостей) та рішень (системою вимог до ділових процесів підтримки елементарних властивостей і термінальних сервісів їх реалізації), поєднані спеціалізованою порівневою моделлю варіабельності композиції. Процес її побудови й використання подано композицією функцій управління варіабельністю цього сімейства.

4. Запропоновано технологічну схему процесу ітеративної побудови варіа-

бельного композитного сервісу згідно з заданими вимогами. Схема доповнює фазу автоматичного композивання компонентних сервісів початковою фазою автоматизованого формування варіабельних вимог до композиції та завершальною фазою прийняття рішень щодо доцільності й способів її еволюції та адаптації.

5. Виконана інтеграція дозволяє поєднати переваги використаних підходів, релевантні для цільових ситуацій використання композиції.

6. Запропоновані гібридні моделі адаптивної композиції Web-сервісів та процесу її побудови й (одночасного) використання в типових ситуаціях споживання її функцій складають підґрунтя подальших досліджень автора:

– доопрацювання створених моделей для сервісів семантичного вебу;

– дослідження можливостей опису цілей адаптації за допомогою дескриптивної логіки;

– розвитку динамічної моделі властивостей нефункціональними вимогами до композиції та її компонентів.

1. *Андон П., Дерещкий В.* Проблеми побудови сервіс-орієнтованих прикладних інформаційних систем в semantic web середовищі на основі агентного підходу // Проблеми програмування. – 2006. – N 2–3. – С. 493–502.
2. *Bouguettaya A., Sheng Q., Daniel F.* Web Services Foundations Springer Science & Business Media, 4 сент. 2013 – 758 p.
3. *Zeginis C., Plexousakis D.* Web Service Adaptation: State of the art and Research Challenges. Technical Report 410 ICS-FORTH, October 2010. – 66 p.
4. *Kazhamiakin R. et al.* Baseline of Adaptation and Monitoring Principles, Techniques, and Methodologies across Functional SBA Layers. Deliverable PO-JRA-1.2.3. V.1 – 35 p.
5. *Bucchiarone A. et al.* Domain Objects for Dynamic and Incremental Service Composition // *Villari M. et al. (Eds.) Proc. ESOC 2014, Manchester, UK – LNCS 8745, 2014. – P. 62–80.*
6. *Bucchiarone A. et al.* Domain Objects for Continuous Context-Aware Adaptation of

- Service-based Systems // Proc. IEEE 20th International Conference on Web Services. – 2013. – P. 571–578.
7. Bertoli P. et al. Control Flow Requirements for Automated Service Composition // 2009 IEEE International Conference on Web Services. – P. 17–23.
 8. Kazhamiakin R. et al. Data-Flow Requirements for Dynamic Service Composition // Proc. IEEE 20th International Conference on Web Services. – 2013. – P. 243–250.
 9. Bertoli P., Pistore M., Traverso P. Automated composition of Web services via planning in asynchronous domains – Artificial Intelligence. – 2010. – N 174. – P. 316–361.
 10. Alférez G.H., Pelechano V. Facing uncertainty in Web service compositions // Int. J. of Services Computing. – April–June 2014. – Vol. 2, N 2. – P. 1–16.
 11. Kapuruge M., Han J., Colman A. Service orchestration as organization: Building multi-tenant service applications in the Cloud. – 2014 – 363 p.
 12. Adzic G. Impact Mapping: Making a big impact with software products and projects. – Provoking Thoughts Limited, 2012. – 72 p.
 13. Slabospitskaya O., Kolesnyk A. The Model for Enhanced Variability Management Process in Software Product Line // In.: Mayr H.C., Kop C., Liddle S., Ginige A. Information Systems: Methods, Models and Applications. Revised selected papers of 4-th International United Information Systems Conference (UNISCON 2012). Yalta, Ukraine, June 2012. – P. 162–171.
 14. UDDI version 3.0.2 – UDDI spec technical committee draft. Technical Report. – OASIS, 2004.
 15. Карпов Ю.Г. MODEL CHECKING. Верификация параллельных и распределенных программных систем. – БХВ-Петербург; 2010. – 560 с.

Одержано 15.03.2015

Про автора:

Слабоспицька Ольга Олександрівна,
кандидат фізико-математичних наук,
старший науковий співробітник.

Місце роботи автора:

Інститут програмних систем
НАН України,
03187, Київ-187,
Проспект Академіка Глушкова, 40.
Тел.: (044) 526 4579.
E-mail: ols.07@mail.ru