

---

doi:<https://doi.org/10.15407/emodel.40.06.061>

УДК 004.891:007.51

**О.Є. Коваленко**, канд. техн. наук

Ін-т проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України  
(Україна, 30164, Київ, вул. Генерала Наумова, 15),

Ін-т проблем математичних машин та систем НАН України  
(Україна, 30187, Київ, просп. Академіка Глушкова, 42,  
тел. +380982143564, e-mail: Kovalenko.O.E@nas.gov.ua)

## Системна інженерія та життєвий цикл систем

Досліджено сучасне бачення та аспекти системної інженерії. Проаналізовано ядро знань з системної інженерії. Описано елементи системної інженерії, їх особливості та взаємозв'язки між ними. Розглянуто життєвий цикл систем у контексті процесів системної інженерії. Показано місце та значення системної архітектури у підтримці процесів системної інженерії. Обґрунтовано методологічний базис індустріальних технологій створення складних організаційно-технічних систем та керування ними, який забезпечує цілісне подання системи на різних рівнях та з різних точок розгляду.

*Ключові слова*: система, системна інженерія, життєвий цикл систем, системна архітектура.

У широкому розумінні термін «система» означає сукупність технічних, природних чи соціальних елементів або їх комбінацію. Іноді це може привести до неоднозначності: наприклад, управління процесом системної інженерії стосується саме управління чи є керуванням розробленою системою? Як і в інших спеціальних дисциплінах, в системній інженерії використовуються терміни, які можуть бути незрозумілими за межами предметної області. Наприклад, у науці про системи, а отже, і в системній інженерії, «відкрита» означає, що система здатна взаємодіяти зі своїм оточенням, на відміну від «закритої» для оточення системи, але в більш широкому інженерному середовищі «відкриту» розуміють як «неприватну», тобто «публічно доступну». У таких випадках у системі знань намагаються уникати неправильного тлумачення, використовуючи альтернативи: наприклад, «управління системою» або «управління проектуванням систем».

Необхідність вирішення проблем побудови найскладніших технічних систем обумовило потребу у виробленні більш систематичних і цілісних

© О.Є. Коваленко, 2018

підходів до їх вирішення. Для систематизації та узагальнення знань з розробки та використання складних людино-машинних систем під егідою Міжнародної ради з системної інженерії (International Council on Systems Engineering (INCOSE)), Комп'ютерної спілки Інституту інженерів з електротехніки та електроніки (Institute of Electrical and Electronics Engineers Computer Society (IEEE-CS)) та Дослідного центру з системної інженерії (Systems Engineering Research Center (SERC)) було створено та підтримується проект «Ядро знань та навчальна програма з розвитку системної інженерії» (Body of Knowledge and Curriculum to Advance Systems Engineering (BKCASE)), одним з напрямків якого є узагальнення та підтримка в актуальному стані ядра (головної частини) знань з системної інженерії у вигляді керівництва до ядра знань з системної інженерії (Guide to the Systems Engineering Body of Knowledge (SEBoK)) [1], а також забезпечення його використання як безкоштовного і відкритого ресурсу з вільним доступом. Принципи та практики, надані у SEBoK сприяють еволюції системної інженерії як дисципліни.

**Ядро знань з системної інженерії SEBoK [1]** позиціонує системну інженерію у ширшому розумінні, тобто система розглядається як фундаментальна частина. Для цього, не намагаючись перевизначити загальну системну термінологію, SEBoK запроваджує два пов'язані визначення, специфічні для системної інженерії:

розроблювана система — це технічна або соціально-технічна система, яка є предметом життєвого циклу (ЖЦ) інженерії систем;

розроблювана система крім внутрішніх зв'язків між елементами включає також зв'язки з інженерними, соціальними або природними системами в одному або кількох визначених середовищах.

Проблеми людини як елемента інженерної системи і зростання функціональності та ролі програмного забезпечення були усвідомлені в 1960-х роках [2,3] і активно досліджувались у період з 1970-х [4] до 1990-х років [5—8]. Результатом усвідомлення людського елемента в технічних системах була переорієнтація з традиційної системної інженерії на «програмні» підходи у системній інженерії. Для традиційної апаратно-орієнтованої системної інженерії характерними є послідовні процеси, заздалегідь задані вимоги, функціонально-ієрархічні архітектури, рішення на основі математичних обґрунтувань та розробка системи за один етап. Підхід до системної інженерії на основі програмних систем зумовлений вимогами, що розвиваються і змінюються, одночасним визначенням вимог та рішень, поєднанням багаторівневої сервісно-орієнтованої та функціонально-ієрархічної архітектур, рішеннями на основі евристик та еволюційною розробкою системи. Зростання ролі програмного забезпечення як критичного

елемента систем зумовило виникнення програмної інженерії (Software Engineering) як дисципліни, тісно пов'язаної з системною інженерією.

За визначенням [1, 9] системна інженерія є міждисциплінарним підходом, що надає засоби для реалізації успішних систем. Вона сфокусована на цілісному та одночасному розумінні потреб зацікавлених сторін, вивченні можливостей, документуванні вимог, а також синтезі, верифікації, валідації та еволюціонуванні рішень при розгляді проблеми в цілому, починаючи від розробки концепції системи до її передачі.

Успішні системи повинні задовольняти потреби своїх споживачів, користувачів та інших зацікавлених сторін. Ключові елементи інженерії систем (рис. 1) включають наступне.

- Принципи та поняття, що характеризують систему як комбінацію елементів, що взаємодіють між собою для досягнення визначених цілей. Система взаємодіє зі своїм середовищем, яке може включати інші системи, користувачів та природне середовище. Елементи, що складають систему, можуть включати апаратне та програмне забезпечення, мікрокод (прошивку), людей, інформацію, методи, засоби, служби та інші підтримуючі елементи.

- Системний інженер — це людина або роль, яка підтримує міждисциплінарний підхід. Системний інженер виявляє і переводить потреби клієнта в специфікації, які можуть бути реалізовані командою розробників системи. Допмагаючи в реалізації успішних систем, системний інженер підтримує множину (набір) процесів ЖЦ, починаючи з ранніх етапів концептуального проектування і далі, протягом всього ЖЦ системи, через її створення, розгортання, використання та утилізацію. Він повинен аналізувати, специфікувати, розробляти та перевіряти систему, щоб забезпечити збалансованість її функціональних, інтерфейсних, робочих, фізичних та інших показників якості та вартості для задоволення потреб зацікавлених сторін. Системний інженер допомагає забезпечити сумісність елементів системи для досягнення цілей і, зрештою, задовольнити потреби користувачів та інших зацікавлених сторін, які придбають та будуть використовувати систему.

Системна інженерія сфокусована на забезпеченні спільної роботи частин системи для досягнення загального призначення і базована на поєднанні практик, що застосовуються у ряді суміжних галузей (зокрема, аерокосмічній та оборонній), які були використані як основа для стандартизованого підходу до управління ЖЦ будь-якої складної системи. Отже, практики системної інженерії на даний час значною мірою ґрунтуються на евристичних методах. Поточні дослідження спрямовані на розвиток теоретичної основи для системної інженерії з врахуванням основних знань із

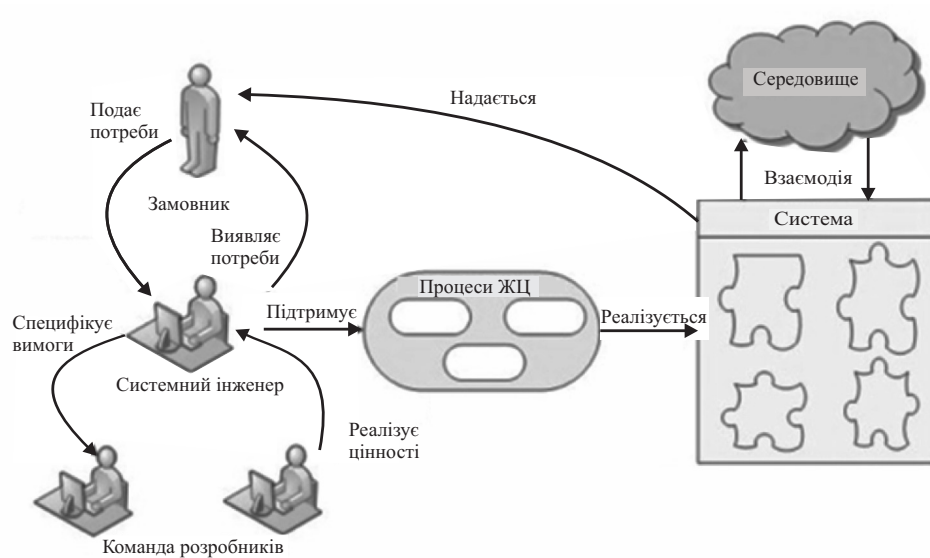


Рис. 1. Ключові елементи системної інженерії [1]

різних джерел. Системна інженерія розвивається у відповідності до постійного підвищення складності систем. Значної мірою ця еволюція пов'язана з моделями та інструментами, орієнтованими на конкретні аспекти системної інженерії, такі як розуміння вимог зацікавлених сторін, подання системних архітектур або моделювання конкретних системних властивостей. Інтеграція між різними дисциплінами, етапами розробки та проектами залишається ключовою проблемою системної інженерії.

Системна інженерія отримала визнання у різних галузях промисловості, науки та державного управління. Однак практика системної інженерії відрізняється залежно від галузей, організацій та типів систем. Взаємне збагачення практик системної інженерії повільно, але неухильно, відбувається у різних галузях діяльності, однак глобальна потреба у спроможностях систем перевершила прогрес у системній інженерії.

Сфера застосування системної інженерії — це не все, що стосується інженерії та управління розробленою системою. Діяльність може бути частиною середовища системної інженерії, за винятком управління специфічною функцією, яка не вважається її частиною. Незважаючи на те що системна інженерія забезпечує реалізацію успішної системи, у випадку, коли певний вид діяльності не входить у сферу охоплення системної інженерії (наприклад, погано кероване та погано організоване виробництво), системна інженерія не може забезпечити успішної реалізації.



Рис. 2. Перетин областей знань інженерії систем, їх реалізації та управління проектами (системами) [1]

Зручним способом визначення охоплюваної сфери діяльності системної інженерії є використання діаграми Венна. На рис. 2 показано взаємозв'язок між системною інженерією, реалізацією системи та управлінням проектами (системами). Такі види діяльності, як аналіз альтернативних методів розробки, тестування та використання, є частиною функцій планування та аналізу системної інженерії. Така діяльність, як замовлення та монтаж виробничих ліній, а також їх використання у виробництві, хоча і є важливими для навколишнього середовища, зрештою залишаються поза межами системної інженерії. Слід зауважити, що інженерна реалізація системи включає також аспекти, пов'язані з виробництвом програмного забезпечення системи, тому програмна інженерія не розглядається як підмножина системної інженерії.

У документі Systems Engineering Vision 2025 [10] INCOSE робить висновок, що системна інженерія готова відігравати важливу роль у деяких глобальних викликах 21-го століття і вже почала змінюватися, щоб відповідати цим викликам, але повинні відбутись більш значущі трансформації, щоб повністю подолати ці виклики. Ключові пункти, взяті з підсумкового розділу [10], визначають атрибути трансформованої дисципліни системної інженерії в майбутньому:

- Відповідність широкому колу областей застосування далеко за межами традиційного застосування в аерокосмічній та оборонній сферах для задоволення зростаючих пошуків системних рішень, що забезпечують основні потреби у глобальному конкурентному середовищі.
- Більш широке застосування соціально-фізичних систем для проведення оцінювання при підтримці прийняття політичних рішень та різних форм відновлення.
- Всебічна комплексна інтеграція вимог з боку різних ринкових, соціальних та екологічних зацікавлених сторін з урахуванням повного ЖЦ та довгострокових ризиків.
- Ключова роль інтеграції для підтримки співпраці, що охоплює різноманітні організаційні та регіональні межі, а також широкий спектр дисциплін.
- Підтримка теоретичним базисом та досконаліми, ґрунтованими на моделях, методами та інструментами, що дозволяє краще зрозуміти найскладніші системи та рішення в умовах невизначеності.
- Покращена освітня інфраструктура, орієнтована на системне мислення та системний аналіз на всіх етапах навчання.
- Підготовка професіональних кадрів, які мають не тільки технічну освіту в предметній області, але й володіють передовими інструментами і методами, необхідними для вирішення поточних системних та інтеграційних завдань.

Деякі з цих напрямків системної інженерії розглядаються в SEBoK. Інші необхідно запровадити та повністю інтегрувати в області знань про системну інженерію у процесі їх розвитку. Ці області знань використовуються для проведення огляду трансформуючих аспектів системної інженерії та інтегруються у всі аспекти SEBoK в процесі удосконалення.

Діяльність з визначення системи забезпечує створення та детальний опис цільової системи (ЦС, system-of-interest, SoI) для задоволення виявленої потреби. Ця діяльність описується як основоположні процеси, що включають визначення системних вимог, системної архітектури, задуму системи та системного аналізу. Визначення архітектури системи може включати розробку моделей логічної та фізичної архітектури. Під час або в кінці будь-якої ітерації виконується аналіз пробілів для забезпечення від-



повідності всіх системних вимог архітектурі та задуму. Діяльність щодо визначення системи спирається на артефакти та рішення, отримані на етапі визначення концепції, в першу чергу, при чіткому формулюванні призначення ЦС, потреб та вимог зацікавлених сторін та попередніх робочих концепцій.

**Організація ЖЦ систем.** Поняття ЖЦ систем почали використовувати при дослідженні соціально-економічних систем у 1950-х роках [11, 12]. Це поняття логічно, структуровано та методично описує процеси систем (продуктів) на кожному етапі їх існування. У роботі [13] зазначено, що ЖЦ розробки систем був запроваджений в 1960-х роках для створення великомасштабних функціональних бізнес-систем. Кількість етапів (фаз) ЖЦ системи та їх назви залежали від галузі застосування та призначення системи.

Термін «життєвий цикл» запозичений з природничих наук і використовується для опису змін, які зазнає окремий організм протягом життя, і взаємодії багатьох організмів з метою розвитку своєї популяції. В системній інженерії ЖЦ використовується для опису повного циклу існування екземпляра ЦС, а також для опису керованої комбінації кількох таких екземплярів, які підтримують спроможності, що забезпечують потреби зацікавлених сторін.

Модель ЖЦ визначає основні етапи, через які проходить специфічна ЦС, починаючи з моменту її започаткування до виводу з експлуатації. Моделі ЖЦ, як правило, реалізуються у проектах розробки та строго узгоджуються з плануванням управління та прийняттям рішень.

Загальна парадигма системної інженерії (рис. 3) визначає загальні цілі зусиль з розробки системи, а саме: розуміння потреб зацікавлених сторін; вибір конкретної потреби, яку необхідно задовольнити; перетворення цієї потреби у систему (продукт або послугу), що забезпечує цю потребу; використання цього продукту або послуги для забезпечення цінностей зацікавлених сторін. Ця парадигма відповідає принципам системного підходу і використовується як базова для області знань SEBoK.

На рис. 3, а, показано структурну декомпозицію системи на ЦС (SoI) в процесі її формування. ЦС1 розділено на основні елементи, які в даному випадку також є системами — ЦС2 і ЦС3, що складаються з системних елементів E1, ..., E4. На рис. 3, б, кожна ЦС має відповідну модель ЖЦ, яка складається з етапів, наповнених процесами. Функція цих процесів полягає у визначенні роботи, яка повинна виконуватися, і пов'язаних з нею артефактів, що будуть створені. У модельно-орієнтованому підході ці артефакти фіксуються в моделі системи, яка представляє собою ЦС. Зауважимо, що деякі вимоги, визначені для задоволення потреб, розподіляються на ранніх етапах ЖЦ для ЦС1, тоді як інші включаються у ЖЦ для ЦС2

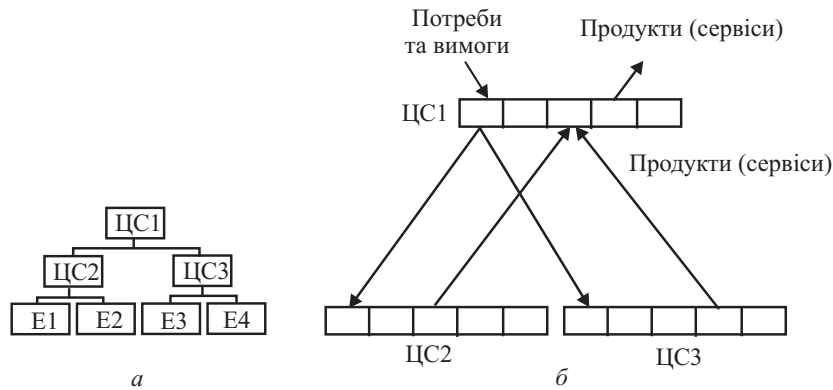


Рис. 3. Загальна парадигма системної інженерії: а — структурна декомпозиція системи; б — ЖЦ (процеси) [1]; Е — системний елемент

або ЦС3. Декомпозиція системи реалізує фундаментальне поняття рекурсії, як визначено в стандарті ISO/IEC/IEEE 15288:2015 [14], тобто положення стандарту будуть повторно застосовані для кожної ЦС. Важливо зазначити, що вимоги можуть бути розподілені по різних елементах системи, інтегрованих на різних етапах ЖЦ будь-якої з трьох ЦС, однак разом вони утворюють цілісну систему. Наприклад, ЦС1 може бути простою машиною, що складається з шасі, двигуна та елементів керування, ЦС2 — вбудованою апаратною системою, а ЦС3 — основним програмним інтерфейсом та системою керування.

Під час виконання процесів ЖЦ системної інженерії потрібно послідовно проводити ітерації між етапами (наприклад, при уточненні визначення системи або при наданні оновлення чи модернізації існуючої системи). Робота, яка виконується в процесах та етапах, може бути виконана паралельно в ЖЦ будь-якої із ЦС, а також у різних ЖЦ.

Цей підхід є фундаментальною основою для розуміння системної інженерії в цілому, а також для застосування системної інженерії до різних типів систем.

**Узагальнена модель ЖЦ систем.** Кожна ЦС має пов'язану з нею модель ЖЦ. Системна інженерія зазвичай синхронізована з різними варіантами таких моделей ЖЦ, які могли б повністю задовольнити потреби зацікавлених сторін. Не існує єдиної моделі ЖЦ, яка підходить до всіх систем і може надати конкретні вказівки для всіх можливих ситуацій проекту. Стандарт ISO/IEC 15288:2015 визначає узагальнену структуру процесів ЖЦ (рис. 4), яка є початковою точкою для найпоширеніших версій заздалегідь визначених, послідовних, еволюційних, ситуативних та паралельних моделей ЖЦ. Кожна така модель визначається як сукупність



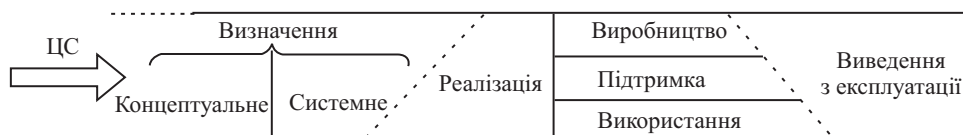


Рис. 4. Узагальнена структура процесів ЖЦ [1]

етапів, в рамках яких виконуються технічні та управлінські заходи. Ці етапи розділені точками прийняття рішень, де основні зацікавлені сторони вирішують, слід перейти до наступного етапу чи залишатись на поточному етапі, припинити чи перемасштабувати пов'язані з ними деталі проектів.

**Концептуальне визначення.** Цей етап починається з рішення ініціатора (особи або організації) інвестувати ресурси в нову або вдосконалену інженерну систему. Ряд зацікавлених сторін погоджуються з необхідністю зміни контексту розроблюваної системи та вивчають, чи можлива розробка нових або модифікованих систем, переваги ЖЦ яких варті вкладання коштів та інших витрат. Діяльність включає: розробку концепції оперування та множин сценаріїв (бізнес-кейсів); визначення ключових зацікавлених сторін та бажаних спроможностей; обговорення вимог між ключовими зацікавленими сторонами та вибір елементів системи, що не будуть розроблятися.

**Системне визначення** починається тоді, коли основні зацікавлені сторони вирішують, що потреби діяльності та вимоги зацікавлених сторін є достатньо чітко визначеними для обґрунтування необхідних варіантів рішення та вартості встановленого ЖЦ. Діяльність включає розробку системних архітектур, визначення та узгодження рівнів системних вимог, розробку планів ЖЦ системних рівнів та проведення системного аналізу для оцінки сумісності та доцільності прийнятого визначення системи. Перехід на стадію реалізації системи зумовлює одноразову або ітераційну розробку.

Слід зазначити, що дані визначення системи описують дії, які виконуються системними інженерами при вирішенні проблем системної інженерії. Існує дуже високий рівень паралелізму між формулюванням проблемної ситуації чи спроможності та описом одного або декількох можливих системних рішень, як це відбувається при системному підході. Інші пов'язані з визначеннями функції включають таке: створення прототипів або фактичну розробку елементів високого ризику для демонстрування можливості реалізації системи; співпрацю з бізнес-аналітиками або проведення аналізу ефективності місії для забезпечення реалізації діяльності; модифікації реалізованих систем для поліпшення їх виробництва, підтримки або використання. Ці заходи зазвичай відбуваються на основі ЖЦ управління розробкою системи, особливо при багатопотоковій розробці.

**Реалізація системи.** Цей етап починається тоді, коли основні зацікавлені сторони вирішують, що архітектура ЦС і аргументи техніко-економічного обґрунтування мають достатньо низькі ризики. Це є обґрунтуванням необхідності визначених ресурсів для розробки та підтримки первинної операційної спроможності або розробки завершеної (повної) операційної спроможності. Діяльність включає: побудову елементів розробки; інтеграцію цих елементів між собою та з нерозроблюваними елементами; верифікацію і валідацію елементів та їх інтеграцію, як це передбачено; підготовку до одночасної діяльності з виробництва, підтримки та утилізації.

**Створення, підтримка та використання системи.** Ці етапи починаються, коли основні зацікавлені сторони вирішують, що доцільність і безпека ЖЦ ЦС мають досить низькі рівні ризику, і це виправдовує використання ресурсів, необхідних для створення, підтримки та утилізації системи протягом очікуваного терміну служби. Тривалість виробництва, підтримки та використання, ймовірно, буде різною. Після завершення виробництва і підтримки користувачі часто продовжують використовувати непідтримувані системи.

**Створення системи** включає виготовлення екземплярів або версій ЦС та потрібних для підтримки запасних частин, а також моніторинг та поліпшення якості продукції; діяльність з приймання продукції або послуг; постійне вдосконалення виробничого процесу. Окрім того, створення системи може включати низькорейтингову початкову продукцію для удосконалення виробничого процесу або сприяння збереженню потенціалу виробництва і подальших збільшень попиту.

**Підтримка системи** включає різні класи технічного обслуговування: коригувальні (для дефектів), адаптивні (для взаємодії з взаємозалежними системами) та удосконалювальні (для підвищення продуктивності, зручності використання). Вона також включає гарячі лінії та відповіді на запити користувачів або надзвичайну підтримку та надання необхідних витратних матеріалів (газу, води, електроенергії тощо). Підтримка системи має кілька сірих меж, а саме межа між невеликими удосконаленнями системи та розробкою нових доповнень, а також межа між переробкою (підтримкою) раніше проведених інкрементів в інкрементній або еволюційній розробці. Зазвичай підтримка продовжується після завершення створення системи.

**Використання системи** включає застосування ЦС операторами, адміністраторами, широким загалом або системами вищих рівнів в ієрархії ЦС. Зазвичай використання продовжується після припинення підтримки системи.

**Виведення системи з експлуатації.** Цей етап часто виконується покроково, оскільки версії системи або її елементів стають застарілими або

нерентабельними для підтримки. Тому вони піддаються вилученню або переробці їх вмісту. Зростання доступних компенсацій робить переорієнтацію системи привабливою альтернативою.

**Процеси системної інженерії** застосовуються для забезпечення коректного прогресування від одного рівня розвитку до наступного деталізованого рівня з використанням контрольованих базових позицій. Ці процеси використовуються для системи, підсистем та системних компонентів, а також для підтримуючих або допоміжних систем, що використовуються при виробництві, експлуатації, навчанні, підтримці та утилізації ЦС. Виконання процесів та заходів технічного менеджменту, таких як дослідження ринку або управління ризиками, може виявити специфічні вимоги, інтерфейси або проектні рішення, які не є оптимальними, і запропонувати зміни для підвищення загальної продуктивності системи, забезпечення економії коштів або дотримання термінів виконання планів. Процес ЖЦ систем являє собою спеціально виділений, функціонально злагоджений процес, який використовується для досягнення цілей і результатів стадій ЖЦ.

Процеси системної інженерії — це сукупність дій, спрямованих на вирішення задач системної інженерії на основі використання холістського підходу, тобто будь-яка система подається як цілісний ансамбль, що поєднує збалансовані аспекти різних видів діяльності: організаційно-господарської; адміністративно-аналітичної; технічної (технологічної).

Організаційно-господарська діяльність поділяється на зовнішню (придбання і постачання) та внутрішню (забезпечення діяльності). Основний зміст функціонування системи полягає у здійсненні процесів технічної діяльності (процесів розробки та використання системи) під управлінням процесів адміністративно-аналітичної діяльності (процесів управління ЖЦ системи) для реалізації свого цільового призначення. Отже, найважливішими типами процесів системної інженерії є такі:

технічні процеси систем;

процеси управління ЖЦ систем (технічного управління).

**Технічні процеси систем** пов'язані зі створенням, впровадженням, використанню та припиненням діяльності систем і визначаються обраною для системи моделлю ЖЦ її розвитку (еволюціонування). Технічні процеси системи забезпечують:

перетворення визначених потреб і побажань зацікавлених осіб в опис збалансованої в рамках ЖЦ сукупності рішень щодо людей, продуктів і процесів;

створення інформації для прийняття рішень;

надання інформації для здійснення подальших технічних зусиль.

**Процеси управління ЖЦ систем** здійснюють контроль над технічними процесами через планування, оцінювання, інформаційне забезпечення, прийняття рішень. Процеси системної інженерії повинні бути здатні до адаптації, під час якої слід визначати та документувати обставини, що впливають на адаптацію. Ці обставини включають (але не обмежують) наступне:

- стабільність і різноманітність середовища функціонування;
- комерційні і (або) експлуатаційні ризики, що стосуються зацікавлених сторін;
- новизну, розміри і складність створюваної системи;
- дату початку та заплановану тривалість застосування;
- питання цілісності, такі як безпека, захист, секретність, зручність застосування, доступність;
- нові (що виникають) технологічні можливості;
- бюджетний профіль і доступні організаційні ресурси;
- готовність систем забезпечення до надання послуг.

Узагальнену модель відношень між процесами розробки систем наведено на рис. 5. Можна виділити такі етапи ЖЦ системи: дослідження; концептуалізація; розробка; реалізація; використання (підтримка); виведення з експлуатації.

В результаті успішного управління ЖЦ визначається:

- модель ЖЦ і сукупність процесів ЖЦ, які будуть використані організацією;
- політика і процедури застосування моделі і процесів ЖЦ;
- політика адаптації типових процесів ЖЦ до потреб окремих проектів;
- показники, які дозволяють контролювати характеристики виконання процесів ЖЦ.

**Системна інженерія на основі модельно-орієнтованого підходу.**

Системні інженери завжди використовували багато типів моделей, включаючи функціональні моделі для підтримки розробки вимог, моделі для аналізу поведінки систем та інші аналітичні моделі для аналізу надійності, безпеки, масової властивості, споживання енергії і вартості. Проте системна інженерія суттєво залежить від артефактів на основі документів для фіксації інформації про розробку, наприклад вимог, документації по інтерфейсу керування та опису архітектури системи. Цю інформацію часто подано у великій кількості різних документів, включаючи текстові, неформальні малюнки та електронні таблиці. Такий документований підхід до системної інженерії має значні недоліки, а саме недостатню точність, невідповідність одного артефакту іншому, труднощі щодо збереження та повторного використання інформації.

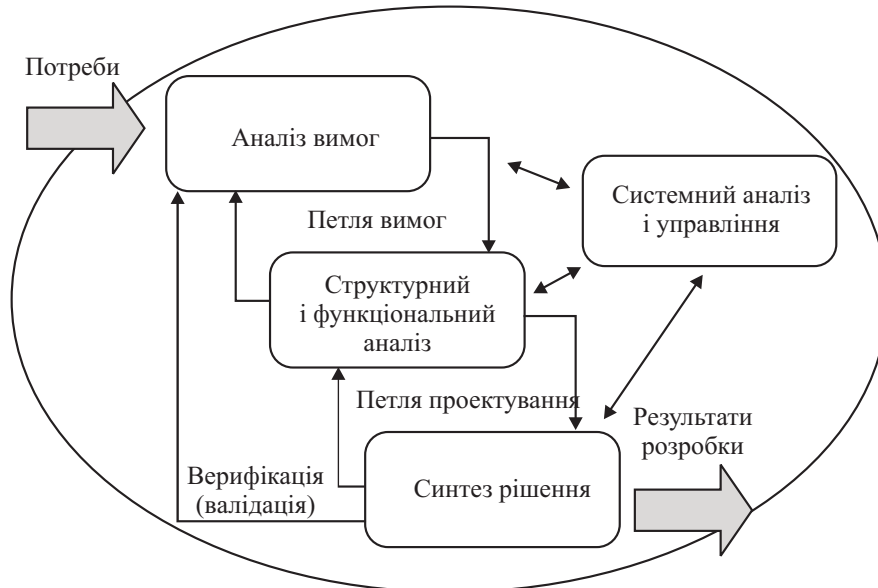


Рис. 5. Модель розробки системи

Модельно-орієнтована системна інженерія ((МОСІ), model-based systems engineering (MBSE)) — це формалізоване застосування моделювання для підтримки діяльності стосовно системних вимог, проектування, аналізу, верифікації та валідації, починаючи з фази концептуального проектування і продовжуючи далі у процесі розробки та на наступних етапах ЖЦ [15]. Особливість підходу МОСІ полягає у тому, що модель є основним артефактом процесу проектування системи. Основна увага при розробці, управлінні та налаштуванні моделі системи є перехід від традиційного підходу, базованого на документах, до системної інженерії, де акцент зроблено на документації про виробництво та налаштування системи. Використовуючи модель системи як основний артефакт, МОСІ надає потенціал для підвищення якості продукції, розширення повторного використання артефактів моделювання системи та покращення зв'язку між командами розробників. Це, в свою чергу, дозволяє зменшити час і витрати на інтеграцію та тестування системи, а також суттєво знизити вартість, ризики та спростити планування при створенні системи.

МОСІ включає різноманітний набір описових та аналітичних моделей, які можуть бути застосовані протягом всього ЖЦ, а також моделей системи систем (SoS) та моделей компонентів. Типові моделі можуть включати описові моделі архітектури системи, які використовуються для ви-

значення та розробки системи, а також аналітичні моделі для аналізу продуктивності системи, фізичних та інших характеристик якості, таких як надійність, ремонтпридатність, безпека та вартість.

Термін MBSE, тобто МОСІ, використано у роботі [16], де наведено формальну систему на основі станів для аналізу систем стосовно їх вхідних (вихідних) характеристик та функцій цінності для оцінки корисності технологічно незалежних та технологічно залежних систем. Моделювання широко застосовувалося у промисловості для забезпечення високоякісного аналізу ефективності складних систем. Стандарт визначення інтеграції для функціонального моделювання [17] було введено у 1990-х роках для підтримки базового функціонального моделювання. Модельний формалізм, що називається блок-схема розширеного функціонального потоку, використовувався для моделювання різних типів систем. Група з об'єктного управління (OMG) розробила концепцію модельно-керованої архітектури (МКА, Model Driven Architecture (MDA)) [18], у якій використано стандартний підхід до моделювання. Мову моделювання систем (OMG SysML) [19] було прийнято у 2006 р. як мову моделювання систем загального призначення. Крім того, у 2008 р. OMG прийняла уніфікований профіль DoDAF та MODAF (UPDM) [20] для підтримки моделювання організацій.

Модельно-орієнтований підхід використовується при побудові різних типів складних людино-машинних систем, зокрема для систем ситуаційного управління [21]. Моделі для систем ситуаційного управління крім технічних та організаційних підсистем описують також сценарії, процедури узгодження, проектну діяльність, шаблони тощо.

**Системна інженерія та системна архітектура.** Системна інженерія [1] передбачає глибоку інтеграцію та узгоджену взаємодію (конвергенцію) технічної, соціально-економічної складових та середовища використання складної системи. Поєднання цих складових забезпечується побудовою та підтримкою архітектури складної системи. Системна архітектура є концептуальною моделлю, яка визначає структурну, поведінкову та інші точки зору на систему. Опис архітектури — це формальний опис та подання системи, організовані так, щоб підтримувати обґрунтування рішень щодо структури та поведінки системи. Стандарт ISO/IEC/IEEE 15288 визначає архітектуру системи, як концептуальну її організацію, втілену в елементах, їх відношеннях між собою і з середовищем, а також визначає принципи, які є основою для прийняття рішень щодо її проектування та еволюціонування.

Архітектура об'єкта, як визначено стандартом ISO/IEC/IEEE 42010 [22], є фундаментальною концепцією або властивостями об'єкта в його



середовищі, втіленими в його елементах, відношеннях та принципах його проектування та еволюціонування. Архітектура відображає:

основні характеристики проблеми та області застосування її рішення з можливими альтернативами, повне рішення включає цільовий об'єкт і сутності, що його підтримують;

координативні дані для процесів, що підтримують ЖЦ рішення, пов'язаного з архітектурою;

інтереси зацікавлених сторін стосовно об'єкта розробки архітектури у формальному вигляді;

припущення стосовно середовища кожного системного рішення для охоплення ЖЦ рішення — операційні процеси, а саме природні, людські та технічні фактори, що взаємодіють з кожною системою; застосовані до них функціональні та нефункціональні обмеження, наприклад як у DLOD (United Kingdom Ministry of Defence Lines of Development), PESTEL (Political, Economic, Social, Technical, Environmental, Legal (Business Evaluation)), DOTPMLFI (Doctrine, Organization, Training, Material, Leadership & Education, Personnel, Facilities and Interoperability/Information) тощо.

Архітектура на рівні діяльності визначає, як група людей або організацій може співпрацювати в роботі з множиною архітектур зі спільним корпоративним баченням і як можна виконувати заходи щодо мотивації діяльності та способів управління проектами. Це забезпечує розуміння ландшафту архітектури разом з робочим середовищем, бібліотеками та сховищами в організації.

Архітектура на рівні спроможностей програми (проекту) охоплює бібліотеки, сховища, портфелі та дії, що використовуються в програмі або проекті спроможностей, який асоціюється з будь-якою архітектурою всередині організації.

*Архітектурний підхід* створює передумови та орієнтири для забезпечення життєздатності організацій і пов'язаних з ними заходів як в організації, так і в окремих проектах. Архітектурний підхід включає такі категорії та їх складові:

*Архітектура організації :*

- організація або стратегічний масштаб;
- мотиваційні дані;
- довідкові бібліотеки;
- репозиторії корпоративної архітектури;
- міграційний план для трансформації;
- портфелі активів;
- політика архітектури;
- архітектурна діяльність.



*Системна архітектура:*

масштаб програми (проекту);  
мотиваційні дані;  
довідкові бібліотеки;  
репозиторії архітектури;  
міграційний план;  
портфелі активів;  
план управління архітектурою;  
проектна архітектурна діяльність.

*Основи архітектурного підходу:*

архітектурні принципи;  
спроможності, засоби, навички та компетенції (інструменти, дисципліни та спеціальності);  
зразки архітектури та побудови архітектур;  
активи — готові для поставки будівельні блоки;  
мотиваційні дані для побудови архітектури — політика та права, контракти, засоби узгодження, готовність та моделі зрілості.

Управління спроможностями здійснюється в масштабі всієї організації, а також на рівні проекту. Опис артефакту спрямовується організаціями та проектами. Архітектурні описи зазвичай відображають набір пов'язаних проблем і розробляються для конкретних зацікавлених сторін.

Опис архітектури включає в себе один або декілька архітектурних видів (architecture views). Архітектурний вид (або просто вид) стосується одного або кількох інтересів зацікавлених сторін щодо цільової системи.

Види (проекції) — це ідеальний механізм для цілеспрямованої передачі інформації про певні проблеми. Вид — це точка зору (viewpoint), яка визначає поняття, моделі, методи аналізу та візуалізацію. Вид відображає архітектуру ЦС відповідно до архітектурної точки зору (architecture viewpoint). Різні види архітектур можна розглядати відповідно до їх цілей, областей застосування та ролей в ЖЦ організації та архітектури. Наприклад, організація може визначити такі види архітектур:

загальнодоступна архітектура, що детально описує майбутню ситуацію (цей опис зазвичай охоплює кілька програм);

архітектура, яка буде використовуватися як еталонна за спроможностями (програмами) або як архітектура в межах домену (області) застосування;

архітектура, що обмежується рамками окремого проекту, який стосується впровадження рішень.

Точка зору охоплює одну або декілька проблем. Зацікавленість може бути сформована більш ніж однією точкою зору. Точка зору встановлює домовленості про визначення та оцінки видів для спрямування проблем,

які охоплено цією точкою зору. Домовленості про точку зору можуть включати мови, нотації (позначки), типи моделі, правила розробки та (або) методи моделювання та інші операції у видах.

Архітектура системи складається з компонентів системи та вже розроблених систем, які будуть функціонувати разом при реалізації загальної системи. Розроблено різні формальні мови для опису архітектури системи, які називаються мовами опису архітектури (ADL), зокрема такі: Functional Flow Block Diagram (FFBD) [23]; Integration Definition for Functional Modeling (IDEF0) [17]; Object-Process Methodology (OPM) [24]; Systems Modeling Language (SysML) [19]; Unified Profile for United States Department of Defense Architecture Framework (DoDAF) and United Kingdom Ministry of Defense Architecture Framework (MODAF) [20]; OWL 2 Web Ontology Language (OWL 2) [25].

**Життєвий цикл архітектури.** Архітектура — це динамічна сутність, яка орієнтує процеси ЖЦ, описані у стандарті ISO/IEC/IEEE 15288, для системи з визначеною архітектурою. Архітектура має власний ЖЦ (побудова, використання та завершення використання, коли ця архітектура більше не застосовується або не задовольняє вимогам). Усвідомлення динамічної сутності архітектури обумовило початок розробки нових архітектурних стандартів для розширення контексту застосування архітектурного підходу у системній інженерії. Зокрема, разом з роботою по оновленню стандарту опису архітектури систем ISO/IEC/IEEE 42010 [22] ведуться та плануються роботи по створенню нових стандартів ISO [26]: 42020 — архітектурні процеси (випуск очікується у 2018 р.); 42030 — модель оцінки архітектури (випуск очікується у 2019 р.); 42040 — архітектурні методи (планується); 42050 — архітектурні засоби (планується).

Процеси (або види діяльності) підтримують цільовий об'єкт протягом його ЖЦ, тобто підтримують будь-яку діяльність, необхідну для того, щоб зробити цей об'єкт життєздатним протягом його ЖЦ. При безпосередній асоціації з системою ЖЦ архітектури відображає весь ЖЦ системи від визначення концепції до виведення з експлуатації. Архітектура забезпечує технічні домовленості власникам та розробникам систем на основі архітектурного плану, формування варіантів системи і підсистем цільового призначення та пов'язаних з ними допоміжних систем. Ці домовленості включають критичний шлях від початку розробки до реалізації численних інкрементів, керованих відповідними версіями планів інженерного управління системою. Іноді архітектура може відображати різні очікування, безпосередньо не пов'язані з окремою системою або її ЖЦ. Приклади таких очікувань наступні.

*Архітектура, створена для ідентифікації системи (систем),* описує проблемний простір, щоб забезпечити вирішення проблем, які стосуються

зацікавлених сторін. У такому випадку розробляються лише точки зору на діяльність (операції) та точки зору на спроможності. Вони використовуються для оновлення доктрини, робочих процесів або для придбання та управління системами або послугами. ЖЦ архітектури починається, коли починається аналіз проблеми, і завершується при найбільшому числі перетинів між просторами проблем та їх рішень.

*Архітектура, створена для охоплення кількох проектів*, які виконуються одночасно протягом певного періоду часу. Її можна назвати загальною, а набір проектів розглядати як програму. ЖЦ такої архітектури починається з початком програми і закінчується останнім проектом.

*Архітектура, створена для охоплення декількох систем (продуктів)*, які працюють одночасно протягом певного періоду часу. До цього варіанту належать виробничі лінії, сімейства систем та системи систем. Архітектура забезпечує загальне визначення, яке зазвичай вдосконалюється окремими архітектурами систем (продуктів).

*Архітектура, створена для покриття декількох проектів*, які виконуються послідовно, коли це можливо, протягом певного періоду часу. У цьому випадку архітектура надає план дій щодо трансформації, включаючи розробку та (або) заміну систем (продуктів), для виконання архітектурних завдань на визначеному проміжку часу.

Ці приклади підкреслюють необхідність налаштування архітектурних середовищ, дій та результатів на те, щоб зробити їх придатними для використання за призначенням. Таке налаштування залежить від організації підприємства та складності проблеми і її рішення може бути застосовано до різних планів та діяльності на архітектурному об'єкті.

## **Висновки**

Розробка сучасних складних людино-машинних систем різного призначення вимагає індустріального підходу для забезпечення контрольованості процесів проектування, випуску, застосування та виведення з експлуатації таких систем. Проведений аналіз підходів до побудови і управління складними організаційно-технічними системами показав, що методологічним базисом при цьому є архітектурний підхід та використання концепції ЖЦ при розгляді складних систем.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. *SEBoK*. Guide to the Systems Engineering Body of Knowledge. 2017. Vol. 1.9. [Електронний ресурс] Режим доступу: [https://www.sebokwiki.org/wiki/Guide\\_to\\_the\\_Systems\\_Engineering\\_Body\\_of\\_Knowledge\\_\(SEBoK\)](https://www.sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK)).
2. *Goode H., Machol R.* Systems Engineering: An Introduction to the Design of Large-Scale Systems. NY: McGraw-Hill, 1957, 551 p.
3. *Warfield J.N.* Systems Engineering. Pennsylvania State University. Navy Department, Ordnance Research Laboratory, Report NO. Nord 7958-307, 1955, 31 p. [Електронний ресурс] Режим доступу: <http://warfield.gmu.edu/items/show/178>.
4. *Bertalanffy L. von.* General System Theory: Foundations, Development, Applications. NY : George Braziller, 1968, 289 p.
5. *Wymore A.W.* A Mathematical Theory of Systems Engineering: The Elements. Huntington, NY: Robert E. Krieger, 1967, 353 p.
6. *Checkland, P.B.* Systems Thinking, Systems Practice. Hoboken, NJ: Wiley, 1981, 330 p.
7. *Yourdon E.* Managing the Systems Life Cycle: A Software Development Methodology Overview. NY: Yourdon Press, 1988, 158 p.
8. *Rechtin E., Maier M.* The Art of Systems Architecting (Systems Engineering). Boca Raton, FL: CRC Press, 1997, 288 p.
9. *INCOSE*. 2015. Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, Fourth Edition. International Council on Systems Engineering (INCOSE). Hoboken, NJ: John Wiley & Sons Inc., 2015, 305 p.
10. *INCOSE*. 2014. A World in Motion: Systems Engineering Vision 2025, July, 2014. International Council on Systems Engineering (INCOSE), 49 p. [Електронний ресурс] Режим доступу: <https://www.incose.org/docs/default-source/aboutse/se-vision-2025.pdf>.
11. *Barton S.G.* The Life Cycle and Buying Patterns // Consumer Behavior, Lincoln H. Clark, ed., Vol. II. NY: New York University Press, 1955, p. 53-112.
12. *Wells W.D., Gubar G.* Life cycle concept in marketing research // Journal of Marketing Research, 1966, Vol. 101, p. 355—363.
13. *Elliott G.* Global Business Information Technology: an Integrated Systems Approach. Boston: Addison-Wesley, 2004, 520 p.
14. *ISO/IEC/IEEE 15288:2015* Systems and software engineering — System life cycle processes. [Електронний ресурс] Режим доступу: <https://www.iso.org/standard/63711.html>.
15. *Estefan J.A.* Survey of Model-Based Systems Engineering (MBSE) Methodologies. May 23, 2008, 70 p. [Електронний ресурс] Режим доступу: [http://www.omg.sysml.org/MBSE\\_Methodology\\_Survey\\_RevB.pdf](http://www.omg.sysml.org/MBSE_Methodology_Survey_RevB.pdf)
16. *Wymore A.W.* Model-Based Systems Engineering. Boca Raton, FL: CRC Press, 1993, 710 p.
17. *IDEF0*. Standard for Integration Definition for Function Modeling. Draft Federal Information Processing Standards, Publication 183, December 21, 1993, 128 p. [Електронний ресурс] Режим доступу: <http://www.idef.com/wp-content/uploads/2016/02/idef0.pdf>.
18. *OMG MDA*. Object Management Group (OMG). Model-Driven Architecture (MDA) Guide, V1.01, June 12, 2003 [Електронний ресурс] Режим доступу: <https://www.omg.org/mda/>.
19. *OMG SysML*. Object Management Group (OMG). OMG Systems Modeling Language (OMG SysML™). V1.4, OMG document number formal/2015-06-03, September 2015, [Електронний ресурс] Режим доступу: <https://www.omg.org/spec/SysML/1.4/>
20. *OMG UPDM*. Object Management Group (OMG) Unified Profile for DoDAF/MODAF (UPDM). OMG document number formal/2013-08-04, August 2013 [Електронний ресурс] Режим доступу: <https://www.omg.org/spec/UPDM/2.1/>

21. *Vlasova T., Kovalenko O., Kosolapov V.* Organizational-Information Technology for Providing and Decisions Making in Situational Management // 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET 2018. Proceedings, Vol. 10, April 2018, p. 152—157. DOI: 10.1109/TCSET.2018.8336176.
22. *ISO/IEC/IEEE 42010:2011 Systems and software engineering — Architecture description* [Електронний ресурс] Режим доступу: <https://www.iso.org/standard/50508.html>.
23. *Oliver D.W., Kelliher T.P., Keegan Jr.J.G.* Engineering Complex Systems with Models and Objects. NY: McGraw Hill, 1997, 325 p.
24. *ISO/PAS 19450:2015 Automation systems and integration — Object-Process Methodology* [Електронний ресурс] Режим доступу: <https://www.iso.org/standard/62274.html>.
25. *OWL 2. Web ontology language. Document Overview (Second Edition), 2012* [Електронний ресурс] Режим доступу: <https://www.w3.org/TR/owl2-overview/>.
26. *Martin J.N.* Overview of Emerging ISO Standards on Architecture, 2018 [Електронний ресурс] Режим доступу: <http://gsaw.org/wp-content/uploads/2018/03/2018s05martin.pdf>

Отримано 01.10.18

#### REFERENCES

1. SEBoK. (2017), Guide to the Systems Engineering Body of Knowledge, v. 1.9, released 17 Novembe. 2017, available at: [https://www.sebokwiki.org/wiki/Guide\\_to\\_the\\_Systems\\_Engineering\\_Body\\_of\\_Knowledge\\_\(SEBoK\)](https://www.sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK)).
2. Goode, H. and Machol, R. (1957), *Systems Engineering: An Introduction to the Design of Large-Scale Systems*, McGraw-Hill, NY.
3. Warfield, J.N. (1955), *Systems Engineering*, *Pennsylvania State University, Navy Department, Ordnance Research Laboratory*, Report NO, Nord 7958-307, available at: <http://warfield.gmu.edu/items/show/178>.
4. Bertalanffy, L. von. (1968), *General System Theory: Foundations, Development, Applications*, George Braziller, NY.
5. Wymore, A.W. (1967), *A Mathematical Theory of Systems Engineering*, The Elements, Robert, E. Krieger, NY.
6. Checkland, P.B. (1981), *Systems Thinking, Systems Practice*, Wiley, NJ.
7. Yourdon, E. (1988), *Managing the Systems Life Cycle: A Software Development Methodology Overview*, Yourdon Press, NY.
8. Rechtin, E. and Maier, M. (1997), *The Art of Systems Architecting (Systems Engineering)*, CRC Press.
9. INCOSE. (2015), *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, Fourth Edition. International Council on Systems Engineering (INCOSE), John Wiley & Sons Inc, NJ.
10. INCOSE. (2014), *A World in Motion: Systems Engineering Vision 2025*, July, 2014, *International Council on Systems Engineering (INCOSE)*, available at: <https://www.incose.org/docs/default-source/aboutse/se-vision-2025.pdf>.
11. Barton, S.G. (1955), “The Life Cycle and Buying Patterns”, *Consumer Behavior*, by Lincoln, H. Clark, *New York University Press*, Vol. 2, pp. 53-112.
12. Wells, W. D. and Gubar, G. (1966), “Life cycle concept in marketing research”, *Journal of Marketing Research*, Vol. 101, pp. 355-363.
13. Elliott, G. (2004), *Global Business Information Technology: an Integrated Systems Approach*, Addison-Wesley, Boston.

14. ISO/IEC/IEEE 15288:2015. (2015), Systems and software engineering – System life cycle processes, available at: <https://www.iso.org/standard/63711.html>
15. Estefan, J.A. (2008), Survey of Model-Based Systems Engineering (MBSE) Methodologies, May 23, 2008, available at: [http://www.omg.sysml.org/MBSE\\_Methodology\\_Survey\\_RevB.pdf](http://www.omg.sysml.org/MBSE_Methodology_Survey_RevB.pdf).
16. Wymore, A.W. (1993), Model-Based Systems Engineering, CRC Press.
17. IDEF0. (1993), Standard for Integration Definition for Function Modeling, *Draft Federal Information Processing Standards*, Publication 183, available at: <http://www.idef.com/wp-content/uploads/2016/02/idef0.pdf>.
18. OMG MDA. (2003), Object Management Group (OMG), *Model-Driven Architecture (MDA) Guide*, Vol. 1.01, available at: <https://www.omg.org/mda/>.
19. OMG SysML. (2015), Object Management Group (OMG), *OMG Systems Modeling Language (OMG SysML™)*, Vol. 1.4, available at: <https://www.omg.org/spec/SysML/1.4/>.
20. OMG UPDM. (2013), Object Management Group (OMG) Unified Profile for DoDAF/MODAF (UPDM), available at: <https://www.omg.org/spec/UPDM/2.1/>.
21. Vlasova, T., Kovalenko, O. and Kosolapov, V. (2018), “Organizational-Information Technology for Providing and Decisions Making in Situational Management”, *14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering*, TCSET 2018, *Proceedings*, Vol. 10, April, 2018, pp. 152-157, DOI: 10.1109/TCSET.2018.8336176.
22. ISO/IEC/IEEE 42010:2011. (2011), Systems and software engineering – Architecture description, available at: <https://www.iso.org/standard/50508.html>.
23. Oliver, D.W., Kelliher, T.P. and Keegan, J.G. (1997), Engineering Complex Systems with Models and Objects, McGraw Hill, NY.
24. ISO/PAS 19450:2015. (2015), Automation systems and integration – Object-Process Methodology, available at: <https://www.iso.org/standard/62274.html>.
25. OWL 2. (2012), Web ontology language. *Document Overview* (Second Edition), available at: <https://www.w3.org/TR/owl2-overview/>.
26. Martin, J.N. (2018), Overview of Emerging ISO Standards on Architecture, available at: <http://gsaw.org/wp-content/uploads/2018/03/2018s05martin.pdf>.

Received 01.10.18

*А.Е. Коваленко*

#### СИСТЕМНАЯ ИНЖЕНЕРИЯ И ЖИЗНЕННЫЙ ЦИКЛ СИСТЕМ

Исследовано современное видение и аспекты системной инженерии. Проведен анализ ядра знаний по системной инженерии. Описаны элементы системной инженерии, их особенности и взаимосвязи между ними. Рассмотрен жизненный цикл систем в контексте процессов системной инженерии. Показано место и значение системной архитектуры в поддержке процессов системной инженерии. Обоснован методологический базис промышленных технологий создания сложных организационно-технических систем и управления ими, который обеспечивает целостное представление системы на различных уровнях и с разных точек рассмотрения.

*Ключевые слова*: система, системная инженерия, жизненный цикл систем, системная архитектура.



*O.E. Kovalenko*

#### SYSTEMS ENGINEERING AND SYSTEMS LIFE CYCLE

The modern vision and aspects of system engineering are investigated. The analysis of the systems engineering body of knowledge has been carried out. The elements of system engineering are described, their features are highlighted and the interrelations between them are described. The life cycle of systems in the context of processes of system engineering is considered. The place and significance of the system architecture in support of processes of system engineering are showed. The methodological basis of industrial technologies for the creation and management of complex organizational and technical systems is substantiated. They provide a holistic representation of the system at different presentation levels and from different points of view.

*Key words: system, systems engineering, system life cycle, system architecture.*

*КОВАЛЕНКО Олексій Єпифанович, канд. техн. наук, доцент, ст. наук, співроб. Ін-ту проблем математичних машин та систем НАН України, докторант Ін-ту проблем моделювання в енергетиці ім. Г.С. Пухова НАН України. У 1984 р. закінчив Київський політехнічний ін-т. Область наукових досліджень — інформаційні технології, засновані на використанні моделей знань, системи ситуаційного управління, онтології, мультиагентні системи.*