

ЗАСТОСУВАННЯ РОЗПОДІЛЕНИХ ОБЧИСЛЕНЬ У ДОДРУКАРСЬКІЙ ПІДГОТОВЦІ ВИДАНЬ

© Нерода Т. В., Олійник Р. В., 2015

Досліджено напрями оптимізації додрукарської підготовки видань у поліграфічно-орієнтованій мережевій інфраструктурі.

Викладено концептуальні методи підготовки багатоавторного видання у середовищі комп'ютерно-видавничої системи, які надають авторам безпосередній обмежений одночасний доступ до виділеної персоналізованої області, визначеної у структурі електронної публікації.

Ключові слова: комп'ютерно-видавничі системи, область доступу, прикладний ресурс, інтерфейс як сервіс, комунікаційні середовища, хмаринна інфраструктура.

APPLICATION OF DISTRIBUTED COMPUTING IN PRE-PRESS PREPARATION OF EDITIONS

© Neroda T., Oliynyk R., 2015

Researched directions of optimization of pre-press preparation of editions in the printing oriented network infrastructure.

Outlined conceptual methods of preparation multi-author's editions in the environment of computer publishing system that provide for authors direct limited and simultaneous access to a dedicated personalized area identified in the structure of electronic publication.

Key words: desktop publishing systems, access area, applied resource, interface as service, communication environments, cloud infrastructure.

Вступ

На сучасному етапі розвитку народного господарства одним з найважливіших завдань є створення спеціалізованих програмних засобів для забезпечення підвищення продуктивності праці, поліпшення якості виробленої продукції, зменшення собівартості й термінів виготовлення. Розвиток технологій, якісно перетворюючи інформаційний простір, зумовлює необхідність структурної перебудови поліграфічної промисловості з огляду на реорганізацію видавничого процесу та системи поширення продукції, що не могло не вплинути на перерозподіл обов'язків, який віддавна склався між видавцями і друкарнями: процес складання та верстання, виготовлення, художнього оформлення і навіть виведення на плівки оригінал-макета передано від поліграфічних підприємств до видавництва [12].

Постановка задачі

Сьогодні широкого застосування набули інтерактивні комп'ютерні системи [2], що тмаксимально автоматизують технологічні етапи підготовки видань до друку [16]. Однак, зважаючи на стрімке зростання номенклатури випуску, зменшення накладів, децентралізацію та особливості збуту продукції, здатної витримати жорсткі умови конкуренції [12], подальшу оптимізацію додрукарських процесів доцільно зосередити, зокрема, на реалізації в наявній поліграфічно-орієнтованій мережевій інфраструктурі конфігурованого механізму розподіленого опрацювання окремих фрагментів поточного виробничого завдання.

Аналіз останніх досліджень та публікацій

Значна частка норм виробітку в додрукарській підготовці видання припадає на верстальника, який, визначаючи структуру модульної системи оригінал-макета [8], здійснює розміщення текстових блоків на шпальті в комплексному поєднанні з ілюстративними матеріалами. Інформаційне наповнення таких об'єктів видання, яке готують окремі користувачі (автори) на базі розрізаних прикладних ресурсів, розмаїтими шляхами потрапляє на робоче місце верстальника й лише тоді він безпосередньо розташовує його в публікації [7].

Відтак, наданий матеріал неодноразово зазнає правки, і в оформленому виданні реципієнт оглядає далеко не перше авторське рішення.

Застосовуваний у наявних моделях технологічної підготовки виробництва [1] такий багатократний опосередкований покроковий зворотний зв'язок з авторами (замовниками) [10] істотно сповільнює розвиток макета; отже, актуальним є впровадження концепції спільного доступу до проекту розроблюваного електронного оригінал-макета, яка дозволить змінити підхід та оптимізувати взаємодію суб'єктів технологічного процесу, що дасть змогу суттєво скоротити життєвий цикл виробництва поліграфічної продукції.

Мета дослідження – розроблення методики проектування багатокористувацького середовища додрукарської підготовки видань з профілюванням прав кінцевих користувачів та визначенням персоналізованих областей доступу в інтерактивній публікації.

Профілювання прав кінцевого користувача надало можливість реалізувати персоналізовану область доступу (ОД) в інтерактивній публікації, зарезервовану відповідно до паспорта видання й модульної системи оригінал-макета, та виділити прикладний ресурс (ПР) для підготовки автором (А) інформаційного наповнення (ІН) об'єкта видання (рис. 1). Межі доступу кожного автора (користувача) лише до відповідної персоналізованої області інтерактивної публікації (П), визначеної структурою видання, встановлюються у користувацькому шарі багатоавторної моделі середовища додрукарської підготовки видань (дуга « \longleftrightarrow »).

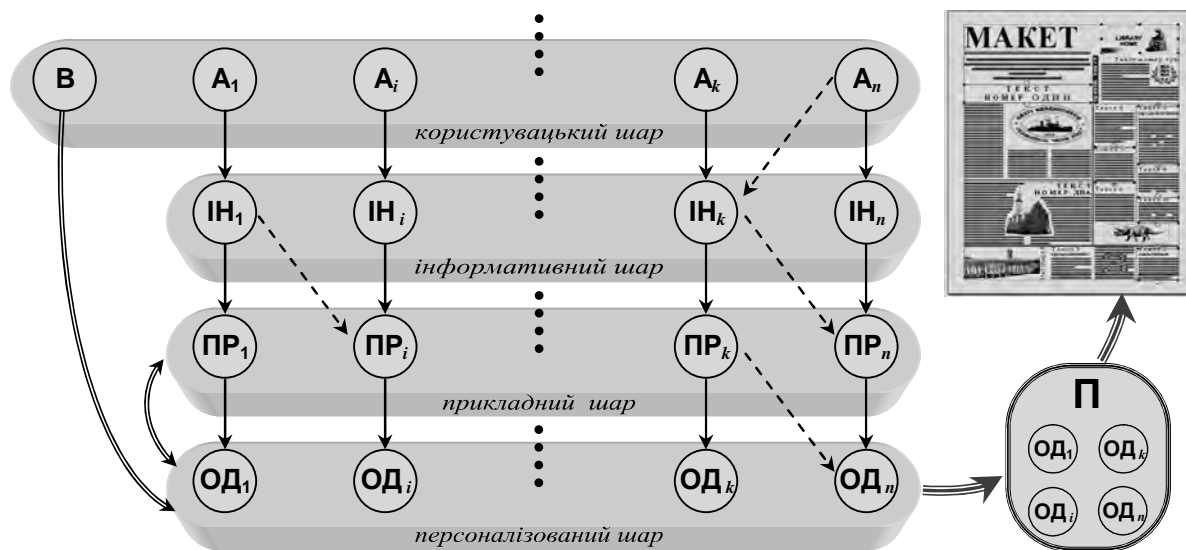


Рис. 1. Ієрархія шарів багатоавторної моделі середовища додрукарської підготовки видань

В інформативному шарі передбачена формалізація за систематизованими ознаками авторського замислу у вигляді сукупності найпростіших складових (поліграфічних елементів) об'єкта видання, які представляють образ наукового поняття опрацьовуваної предметної області [3]. Реалізоване тут інформаційне наповнення підготовлюваної публікації дасть змогу передбачити в наявному математичному апараті вимоги і правила, регламентовані технологічними інструкціями з підготовки відповідних видань вищих рівнів складності, забезпечивши уніфікований набір однотипних об'єктів в усьому виданні з використанням шрифтів і знаків, способом вирівнювання лінійних композицій, застосуванням індексів, лінійок, приставних знаків, нормативно визначених співвідношень між усіма компонентами зображення, основним текстом видання та форматом шпальти згідно з розміткою оригіналу та паспортом видання [4, 16].

Відтак, прикладний шар ініціює прикладні ресурси для опрацювання інформаційного наповнення об'єкта видання, залежно від якого у середовищі прикладного ресурсу моделюються відповідні діалогові засоби коригування інформаційного наповнення [14]: тексту, формул, таблиць, календарів, діаграм, графіки, ілюстрацій, нот тощо. Внесені зміни передаються в автентифіковану область доступу електронної публікації, реалізовану в персоналізованому шарі відповідною задрукуваною площею, де інформаційне наповнення автоматично модифікується згідно з вимогами, зазначеними у паспорті видання [3]. Макетування задрукуваних площ видання, паралельно з їх авторським заповненням, здійснює дизайнер-верстальник [13], який володіє винятковими правами доступу до усіх персоніфікованих областей електронної публікації (дуга « \longleftrightarrow »).

Обумовлену ієрархію шарів багатоавторної моделі середовища додрукарської підготовки видань покладено в основу проекрованої архітектури комп'ютерно-інтегрованого видавничого комплексу. Виконані дослідження [4] показали, що реалізація наведеної концепції спільного мультірівневого доступу до інтерактивної публікації можлива лише за умови застосування в поліграфічно-орієнтованій інфраструктурі конфігурованих мережевих ресурсів, надаваних хмаринними обчисленнями. Зазвичай у типовій хмарі як набір користувацьких сервісів виділяють три головні компоненти: платформа (PaaS), програмне забезпечення (SaaS) та інфраструктура (IaaS); однак виконаний аналіз [15] виявив відсутність у наявних сервісах механізму взаємодії надаваних послуг з використанням прикладного програмованого інтерфейсу (API), через що сервіси зосереджені на одноразовому наданні обмеженого комплексу незв'язаних між собою моделей обслуговування, які вимагають від компетентного користувача самостійно визначати й конфігурувати обчислювальні потреби й автоматизують лише процедури модифікації виділення ресурсів.

Розроблення хмаринної послуги FaaS (інтерфейс як сервіс) та подальше впровадження її в розподілене багатоавторне середовище (рис. 2) як сервіс прямого доступу до платформи та інфраструктури за вимогою користувача (автора, верстальника тощо) дало змогу виправити виявлені недоліки та сформувати свій набір команд для ручного управління потоками даних, зокрема високонавантаженими обчисленнями у багатоавторному середовищі [4, 6].

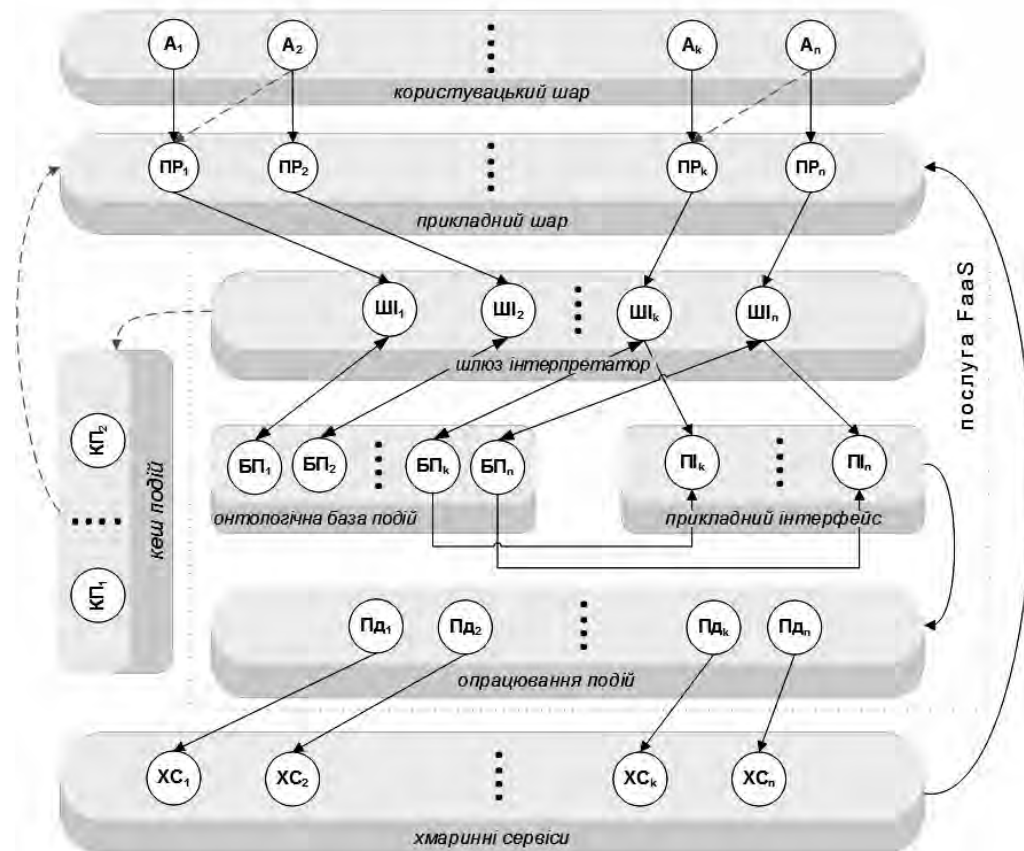


Рис. 2. Інформаційна модель взаємодії компонентів послуги FaaS з багатоавторною KBC

Завдання розроблення відповідного сервісу являє собою проектування користувацького програмованого інтерфейсу, який функціонуватиме разом з системним API. Головною ж відмінністю між надаваним у користування інтерфейсом API та проєктованим є можливість розширення набору функцій у останньому, а також інтеграція у робоче середовище користувача.

Отже, програмований «Інтерфейс як Сервіс» (власне FaaS) виконує роль засобу для прямого запиту на зміну та завантаження відповідних ресурсів хмари за наперед сконфігурованою архітектурою обумовленого фрагмента хмарини для потреб комп'ютерно-інтегрованого видавничого комплексу.

Для конфігурування такої архітектури необхідно забезпечити автоматизоване генерування моделей структур даних для опрацювання подій (Пд), засобів ресурсно-програмної взаємодії у хмаринній мережі та проєктування на їх основі середовища доступу до робочої області фрагмента хмаринної інфраструктури, оскільки тут можливе передавання різнопланової виробничої інформації, а саме відомостей про статус етапу, тип технологічного процесу, стан певної ланки виробництва тощо.

У представленій моделі автори, верстальник та інші оператори комп'ютерно-видавничої системи (КВС) за допомогою прикладного шару (рис. 1) взаємодіють з розподіленим середовищем, що міститься у хмарі, оперуючи при цьому сукупністю інформаційних повідомлень відповідно до обумовленої персоналізованої області доступу користувача (рис. 2), визначеної структурою видання. Область доступу разом з прикладним шаром за спеціалізованим методом динамічно генеруються самою хмарою. Отже, головне завдання FaaS – гнучка й оперативна підтримка поліграфічно-орієнтованого технологічного процесу та безпосередній доступ до управління обчислювальними ресурсами.

Оскільки у комп'ютерно-видавничій системі послідовність етапів автоматизованого моделювання об'єкта видання та задіявані при цьому структури даних переважно сталі, то в проєкті передбачено відповідну онтологічну базу подій (БП), компоненти якої являють собою маркери для сервісів інфраструктури чи платформи на термінове позачергове виконання певного набору функціональних операцій (ХС) хмаринних сервісів, зокрема створення макета публікації у певному форматі вихідних даних. А для отримання гнучкої моделі поведінки проєктованої послуги FaaS у фрагмент хмари багатоавторної КВС введено спеціалізований під потреби КВС прикладний інтерфейс (ПІ), що надаватиме середовище створення командних засобів низькорівневого управління для використання на будь-якій стадії підготовки видання. Необхідно зауважити, що доступ до описуваного прикладного інтерфейсу має користувач з найвищим рівнем автентифікації, оскільки використання FaaS передбачає ігнорування внутрішніх систем безпеки сервісів [5, 11].

Проєктована хмаринна послуга FaaS використовує як математичний апарат наявні пакети інструкцій. Тому для консолідації користувацько-програмної взаємодії застосовується шлюз-інтерпретатор (ШІ), який транслює у машинні команди розроблювані управлінські функції.

Оскільки поліграфічне виробництво працює переважно з тими самими наборами ресурсів та багатокористувацькими середовищами, то доцільно ввести у проєктований фрагмент хмари віртуальну кеш-пам'ять (кеш подій) (КП): головною функціональною особливістю засобу кешування інформації є зберігання готового результату користувацьких команд та налаштованих середовищ автоматизованих редакторів як компонентів КВС для збільшення швидкодії хмари, зменшення часу та витрат на обробку інформації.

Варто зазначити, що команди кеш можуть зберігатися і на боці сервера – класичний випадок, і на боці браузера клієнта (кінцевого користувача) [4]. Усі файли, створювані чи модернізовані у процесі роботи поліграфічного підприємства, зберігаються на серверах компанії хостера, яка забезпечує єдине логічно впорядковане середовище для проєктування ядра програмного забезпечення, баз даних та API [9], а зберігання автономних файлів на стороні клієнта уможливилось завдяки широкому впровадженню у мережеві технології мови розмітки HTML5, яка власне і є головним прототипом взаємодії оператора багатоавторного середовища з хмарою. Недоліком поширеної реалізації засобу кешування можна вважати доволі високу інерційність, особливо в момент оновлення програмного забезпечення.

Натомість, проектована хмаринна інфраструктура повинна забезпечувати високу гнучкість та переналаштування, тому інерційність має дуже негативні наслідки. Щоб позбутися цього недоліку або згладити його, запропоновано, насамперед, профілювати розроблений сервіс FaaS для контролювання кеш-пам'яті, адже тут, на відміну від системного API, передбачено можливість створення нових команд і виконання їх як мікропрограм за вимогою (рис. 2).

Отже, розроблена модель взаємодії потоків даних у хмаринній інфраструктурі уможливорює чіткий розподіл потоків даних на типи: клієнт (кінцевий користувач) – виділений фрагмент хмари; виділений фрагмент хмари – хмаринний сервіс. Також введений сервіс FaaS комутиє інформаційні структури даних усіх компонентів багатокористувацького середовища доступу та поліпшує якісні показники роботи у багатокористувацькому середовищі завдяки оптимізації рутинних команд для управління структурами даних робочих потоків користувацько-програмної взаємодії хмаринних сервісів, а використання засобів кешування надає перевагу не тільки у трафіку, а й у економічних витратах на утримання розгорнутої інфраструктури поліграфічного підприємства.

Проектування засобу маршрутизації розподілених даних забезпечить автоматичне координування виробничих потоків між областю доступу кінцевого користувача та хмаринним середовищем, ґрунтуючись на врахуванні потреб автоматизованого робочого місця (рис. 3).

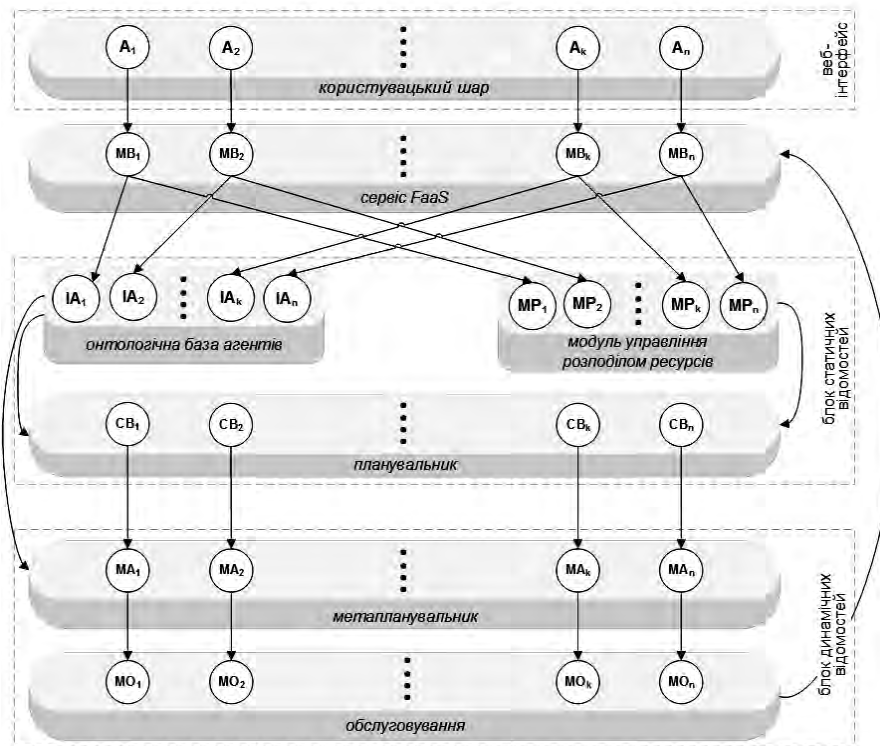


Рис. 3. Розподіл потоків даних у мультиагентному середовищі

Моделі даних, що підтримують раніше обумовлені інформаційні об'єкти та їхні параметри, можливі лише в хмаринній інфраструктурі. Так, блок статичних відомостей являє собою засіб доступу до фрагмента хмаринної інфраструктури, відображаючи веб-інтерфейс як для кінцевого користувача АРМ, так і для зовнішнього фрагмента хмари.

Пізніше отримані відомості трансформуються засобами HTML5 та javascript у мову, прийнятну для опрацювання модулем міжкомпонентної взаємодії FaaS (MB). На цьому етапі FaaS опрацювавши вхідні відомості, формує для модуля управління розподілом ресурсів (MP) активне завдання потреб в обчислювальних ресурсах для компонента “планувальник”. Окрім того, на рівні статичних відомостей розміщено онтологічну базу знань інформаційних агентів (IA), у якій розташовано інформацію про початкові налаштування та параметри затребуваних у процесі роботи хмаринних сервісів.

Головним елементом, яким оперує маршрутизатор потоків даних, є блок специфікацій робочої карти замовлень, що зумовлено потребою комунікації розрізаних фрагментів хмаринних мереж та системи наскрізного управління поліграфічним виробництвом СІР4. Тут вузол поліграфічного продукту містить описову частину технологічного завдання зі складовими частинами виробничого процесу. Наприклад, брошура складається з двох частин – палітурки і книжкового блока. Відповідно буде утворено три вузли: вузол видання, палітурка і книжковий блок. Вузол процесів містить набір правил для окремих робочих кроків процесу виготовлення друкарського продукту або однієї з його частин. До необов'язкових вузлів можна зарахувати вузол груп процесів. Якщо ж на певному етапі виробництва не існує якогось вузла, то його можна також оперативно додати у процесі роботи.

Підготовлені у блоці статичних відомостей дані надходять у блок динамічних відомостей, де відбувається розподіл функціональних обов'язків відповідно до налаштувань планувальника (СВ). На цьому етапі роботи маршрутизатора зі згенерованих на попередньому етапі даних відбувається моделювання сервісу мультиагента (МА) у шарі метапланувальника та здійснюється розподіл інформації, яка опрацьовується блоком динамічних відомостей відповідно до засобів керування інтелектуальних агентів (СВ, МА) (рис. 3) та налаштувань сервісу FaaS (рис. 2). Власне, ці команди утворюють модель розподілу навантаження і є основою для синтезу об'єктної моделі обслуговування (МО).

Отже, процедурний блок динамічної взаємодії розглядається на основі отриманих статичних даних як інтерпретатор, що формує набір агентів для їх подальшого опрацювання хмаринними сервісами, зокрема сервісом FaaS, який сканує потік даних міжрівневої взаємодії. А моделі поведінки інтелектуальних агентів формуються в чіткій послідовності з даними, представленими онтологічною базою знань блока статичних відомостей, та конвертуються у механізм міжсервісної взаємодії як узагальнена модель поведінки метапланувальника.

Описаний механізм роботи програмного маршрутизатора передбачає деяку сукупність домовленостей та інструкцій, зокрема явне передавання повідомлень, активацію об'єктів за пріоритетом, сервер повідомлень, послідовність виклику інтелектуальних агентів, інформацію про обсяг використаних ресурсів хмари тощо. Отже, модель хмаринної інфраструктури ґрунтується на певній множині елементів, до складу якої входять структурні компоненти, специфікації, блоки команд, а також протоколи координації інформаційних ресурсів робочої області [4].

Крім узгодженого функціонування робочих потоків у фрагменті хмари, розроблений маршрутизатор передбачає можливість опрацювання інших файлів, які формують область доступу з хмаринного ресурсу до депозитаріїв з вихідними файлами, що мають відмінне кодування шляху, визначаючи маршрут входження даних. Властивості опрацьовуваного замовлення, елементи конфігурації середовища, а також параметри об'єктної моделі та її атрибути зібрані у блоці динамічної взаємодії та керуються введеним хмаринним сервісом FaaS.

Відтак, застосування багат шарової архітектури програмного забезпечення й розробленого сервісу FaaS у виділеному фрагменті хмаринної інфраструктури істотно полегшує процес розгортання відповідної хмаринної мережі поліграфічного підприємства, забезпечуючи простий спосіб опрацювання XML, SOAP та HTTP-запитів. Виважений підбір компонентів та процедурних блоків розробленого програмного маршрутизатора забезпечує створення коректної моделі області доступу фрагмента хмаринної інфраструктури і є передумовою для повноцінного розгортання віртуального робочого місця та моделювання необхідних прикладних ресурсів.

Висновок

Отже, розроблена модель технологічної підготовки виробництва з програмним маршрутизатором реалізує багат шарову діаграму архітектури послуги FaaS, що дає змогу оперативно проектувати гнучку спеціалізовану структуру розподілених обчислень у додрукарській підготовці

видань. Подальші дослідження доцільно зосередити на інтеграції з наявними системами наскрізного управління поліграфічним виробництвом CIP4 та JDF-workflow як новітніми засобами розгортання та підтримки інформаційного поліграфічного виробництва.

1. Воржева О. В. Автоматизація технологічної підготовки виробництва / О. В. Воржева // *Квалілогія книги: зб. наук. праць.* – 2013. – № 1. – С. 10–14.
2. Дунець Р. Б. Аналіз та синтез топологій комп'ютерних видавничо-поліграфічних систем: монографія. – Львів: НВФ "Українські технології", 2003. – 192 с.
3. Нерода Т. Моделювання структурованих об'єктів при підготовці видань вищих рівнів складності / Т. Нерода // *Автоматизація та комп'ютерно-інтегровані технології у виробництві та освіті: стан, досягнення, перспективи розвитку: матеріали Всеукраїнської науково-практичної Інтернет-конференції, 17–21 березня.* – Черкаси, 2014. – С. 73–75.
4. Нерода Т. В. Проектирование сервис-ориентированных сетевых структур распределённой обработки данных в компьютерно-издательских системах / Т. В. Нерода, Р. В. Олійник // *Механизация, автоматизация, информатизация, телекоммуникация и связь в отраслях производств: монография.* – Новосибирск: СибАК, 2014. – С. 134–153.
5. Олійник Р. В. Аналіз засобів програмного захисту даних в поліграфії / Р. В. Олійник // *Доповіді дев'ятої науково-технічної конференції студентів і аспірантів "Друкарство молоде": Тези доп.* – К.: НТУУ КПІ, 2011. – С. 73–74.
6. Олійник Р. В. Особливості визначення пріоритетів даних у мережевій інфраструктурі поліграфічного підприємства / Р. В. Олійник // *XVI Міжнародна науково-практична конференція з проблем видавничо-поліграфічної галузі: тези доп.* – К., 2013. – С. 64–66.
7. Раттан К. Кроссмедийные системы в полиграфии и издательском деле. Выбор стратегии / К. Раттан. – М.: МГУП, 2007. – 198 с.
8. Сельменська З. М. Модульна сітка в дизайні періодичних видань / З. М. Сельменська, С. М. Комар // *Наукові записки: зб. наук. праць.* – 2014. – № 3. – С. 69–73.
9. Тарнавский Г. А. Облачные вычисления в Интернете: краткий экскурс в центр компьютерного моделирования / Г. А. Тарнавский // *Модел. и анализ информ. систем: сб. науч. трудов.* – М.: ИВМиМГ СО РАН, 2010. – № 2. – С. 112–121.
10. Ткаченко В. Ф. Разработка и исследование автоматизированной системы управления совместной допечатной подготовкой изданий / В. Ф. Ткаченко, И. Н. Нежинская // *Восточно-Европейский журнал передовых технологий.* – 2011. – № 6(2). – С. 55–58.
11. Linthicum D. S. *Cloud Computing and SOA Convergence in Your Enterprise* / D. S. Linthicum. – Boston: Addison-Wesley, 2010. – 265 p.
12. Lutskiy M. Wybrane zagadnienia modelowania i symulacji komputerowej dynamiki maszyn poligraficznych. Lodz: Wydawnictwo politechniki Lodzkiej, 2009. – 120 s.
13. Neroda T. Designation of hierarchical environments of multiauthor's concept of pre-press editions / T. Neroda // *Scientific researches – the theory and experiment '2014: The materials of tenth international scientific and practical conference, Poltava May 26–28, 2014:* – Poltava: Publishing house "InterGrafika", 2014. – Vol. 5. – С. 75–77.
14. Neroda T. Methodology of designing of the specialized application software for desktop publishing: conference proceedings / T. Neroda // *International Conference "Technical sciences: modern issues and development prospects". Conference Proceedings. Scope Academic House, December 10, 2013 – Sheffield, 2013.* – P. 62–64.
15. Oliynyk R. Optimization of multi-architecture network based on cloud computing / R. Oliynyk // *International Conference "Technical sciences: modern issues and development prospects": Conference proceedings.* – Sheffield, UK, 2013. – P. 75–76.
16. Tschichold J. *Ausgewählte Aufsätze über Fragen der Gestalt des Buches und der Typographie.* Basel.: Birkhäuser, 1993. – 214 p.