

УДК 004

Д.Ковтуненко, магістр гр. КН-24М,
Центральноукраїнський національний технічний університет

ДОСЛІДЖЕННЯ ТА ПРИНЦИПИ ПОБУДОВИ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОЇ ІНТЕГРАЦІЇ ІСНУЮЧИХ КОМПОНЕНТ ІТ-СИСТЕМИ З ХМАРНИМИ СЛУЖБАМИ НАСТУПНОГО ПОКОЛІННЯ

У статті розроблено програмне забезпечення, яке призначено для системи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління. Метою розробки є дослідження та принципи побудови системи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління. Об'єктом дослідження є процес інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління. Предметом дослідження є методи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління. Методи дослідження базуються на методах хмарних технологій, методах математичної статистики, методах розробки програмного забезпечення. Результат роботи – програмна реалізація системи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління. В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

інтелектуальна інтеграція, ІТ-система, хмарні служби наступного покоління

Постановка проблеми. Останнім часом значно активізувалися зусилля по реалізації хмарних сервісів, прагнучи відвоювати лідируючі позиції в конкурентів, які раніше прийшли на цей ринок. Cloud Platform надає хмарні інструменти для мобільної розробки, замовленого програмування й модернізації наявних додатків. Хоча Cloud підтримує всі типи хмарних сервісів: IaaS, PaaS і SaaS, – важливе значення надається взаємодії з розроблювачами.

Розходження між інфраструктурою як сервісом і платформою як сервісом можна визначати по-різному. Говорячи із замовниками, я звичайно не акцентую на цьому їхня увага. Однак певна – невелика – різниця зберігається. Ключове слово – автоматизація. У випадку IaaS замовник має можливість більшого контролю, але й залишає за собою більшу відповідальність. Наприклад, він повинен сам установити й зконфігурувати ПЗ, що збирається перенести в хмарну інфраструктуру.

При використанні ж PaaS (у всякому разі, це справедливо у відношенні Cloud Platform) забезпечується високий ступінь автоматизації надання ресурсів, середовища розробки й процесів. Автоматизація розгортання додатків, установки відновлень, резервного копіювання, відновлення й масштабування – все це атрибути платформи як сервісу, які далеко не завжди застосовні до IaaS.

Cloud Platform розрахована на різні категорії клієнтів. Очевидно, що розроблювачі – одна з них, але цільовим сегментом є також ІТ-менеджери, інтегратори додатків і інші користувачі.

Аналіз останніх досліджень і публікацій. При аналізі останніх досліджень і публікацій [1-30] було виявлено певні прогалини у забезпеченні системи інтелектуальної інтеграції існуючих компонент іт-системи з хмарними службами наступного покоління.

Мета й завдання дослідження. Метою роботи є дослідження та принципи побудови системи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.
- Дослідження системи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.
- Програмна реалізація системи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.

Об'єктом дослідження є процес інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.

Предметом дослідження є методи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.

Методи дослідження базуються на методах хмарних технологій, методах математичної статистики, методах розробки програмного забезпечення.

Виклад основного матеріалу. Замовники можуть споживати інфраструктуру як сервіс і платформу як сервіс або з публічної хмари, або зі свого ЦОДу. В останньому випадку надається в оренду все необхідне встаткування й управляємо сервісами точно так само, як у публічній хмарі, тобто замовник одержує ті ж самі ресурси й функції в себе в ЦОДі.

Портфоліо Cloud at Customer включає кілька реалізацій. Наприклад, Cloud Machine надає інфраструктуру й платформу як сервіс. Якщо замовник хоче одержати Exadata як сервіс, ми пропонуємо йому Exadata Cloud Machine і надаємо повний контроль над цими сервісами. За допомогою захищеного шлюзу інженери можуть установлювати необхідні відновлення, але даних клієнтів їм не видні. Вся інформація в нашій хмарі шифрується (при зберіганні, при передачі або в пам'яті), а ключі шифрування відомі тільки замовникові.

Незважаючи на те що в інженерів немає доступу до даних, деякі підприємства не хочуть, щоб до їхнього середовища можна було підключитися ззовні. Із цієї причини ми пропонуємо підтримку так званого відключеного режиму (disconnected mode). Це означає, що регіональний інженер регулярно відвідує замовника для установки відновлень, тобто замість проведення операцій віддалено наші інженери виконують їх на місці. Очевидно, що така послуга обходиться дорожче.

Все залежить від того, що розуміти під галузевою хмарою. Якщо мова йде про специфічний різновид додатків, то ми дуже багато робимо в цьому напрямку. пропонує саме різне програмне забезпечення у вигляді сервісу, призначене для конкретних галузей: телекомунікаційних компаній, роздрібного й фінансового секторів, а також для ряду інших напрямків.

Інший підхід – швидше за все, ви мали на увазі саме його – передбачає наявність окремого ЦОДу для конкретної галузі. У нас є, наприклад, урядова хмара, що відповідає вимогам державних організацій. Є фінансова хмара, побудована з урахуванням відповідних стандартів, зокрема PCI DSS.

Вибір технологій залежить від особливостей законодавства тої або іншої країни. Наприклад, у США державний сектор дуже жорстко регулюється такими нормами, як FedRAMP, тому ми повинні були створити окремий ЦОД для відповідності цим вимогам. Так само ми надходимо й в інших регіонах, де урядове регулювання дуже тверде і є критична маса користувачів, достатня для створення окремої хмари.

Від інших технологій Інтернет речей відрізняється насамперед обсягом даних. Їхня обробка в реальному часі вимагає величезних обчислювальних потужностей.

Для Інтернету речей пропонує різні рішення. Так, розроблений хмарний сервіс IoT Cloud Service, за допомогою якого можна реєструвати різні пристрої, здійснювати безпечні комунікації з ними й управляти двосторонньою взаємодією, а крім того, у реальному часі

обробляти всю вступник від них інформацію. Можна аналізувати ці дані, витягати з них важливі відомості й управляти з їхньою допомогою бізнесами-процесами.

З обліком генеруємого обсягу даних і подій разом з IoT часто використовуються технології Великих Даних, а виходить, крім обробки в реальному часі, потрібна можливість поміщати інформацію в конкретні озера даних. Відповідне середовище дуже важко правильно масштабувати, тоді як еластичність хмари дозволяє починати з малого й нарощувати ресурси в міру необхідності. Ми пропонуємо цілий спектр хмарних сервісів для Великих Даних. Так, Big Data Preparation Cloud Service дозволяє підготувати дані для переносу в середовище Великих Даних, а Big Data Discovery Cloud Service – витягти всю корисну інформацію.

Інтернет речей висуває високі вимоги до інтеграції із внутрішніми системами. Подій відбувається багато, а реагувати на них треба вчасно. Однак для цього необхідно запустити бізнес-процес, що з великим ступенем імовірності буде задіяти різні внутрішні системи, тобто IoT-середовище повинна бути тісно взаємозалежна з ними. Ми надаємо сервіси інтеграції в хмару, щоб можна було запускати необхідні бізнес-процеси при настанні події.

Таким чином, IoT не можна розглядати окремо. Інтернет речей породжує безліч супутніх завдань у таких областях, як керування даними, інтеграція із внутрішніми системами, інформаційна безпека й керування IT.

Про прикордонні обчислення говорять усе більше й більше. Однак можливі різні сценарії. Деякі пристрої, маючи достатні інтелектуальні можливості й необхідну обчислювальну потужність, здійснюють обробку подій в автоматичному режимі, але при цьому потрібно контролювати, які дані вони використовують. Цілісне подання про те, що відбувається на таких пристроях, припускає агрегування й централізацію даних, а для цього необхідно хмара.

Маятник розвитку гойдається туди й назад. Спочатку все поєднується, потім розподіляється, потім знову поєднується й т.д. Хмара – це насамперед централізація. Прикордонні ж обчислення припускають значну децентралізацію. Потреба буде зберігатися й у тому і в іншому. Даних і подій стільки, а потреба в їхній обробці росте так стрімко, що перенос частини обчислень на границю досить бажаний, але хмари при цьому не розсіюються, тому що в одних сценаріях краща децентралізація, в інші – централізація.

Контейнери, безумовно, роблять все більший вплив на те, як замовники розробляють і впроваджують нові додатки. Застосування контейнерів підвищує гнучкість рішення, до того ж додатки ізолюються від платформи, на якій вони будуть функціонувати. Контейнери можна розгорнути у власному ЦОДі, хмарі або будь-якому іншому. Гнучкість – одна із причин того, чому контейнери використовуються всі частіше.

І знову мова йде про волю вибору. У наші дні можливість вибору надзвичайно важлива, оскільки універсального рішення не існує. Я можу виділити три різних підходи до застосування контейнерів. Один з них припускає повний контроль: замовник сам створює контейнери, розгортає їх у хмарі й управляє ними. Ми підтримуємо цю опцію: Docker, Swarm і інші технології, що базуються на Docker, Kubernetes і т.д., можна розгорнути в хмарі й використовувати самостійно. Але такий контроль жадає від замовника більших зусиль.

Інший підхід представлений Container Cloud Service. Замовник сам створює образи контейнерів і розміщає їх у хмарі за допомогою цього сервісу. Хмара відповідає за оркестрацію, розміщення образів, моніторинг і контроль образів Docker. Таким чином, замовникові немає необхідності освоювати всі необхідні технології оркестрації.

Третій розрахований на тих, хто хотів би скористатися всіма перевагами контейнерів, але не знає з технологією. Для них ми пропонуємо Application Container Cloud Service. Замовник може написати застосунок і розгорнути його в контейнерному середовищі, при цьому йому не прийдеться мати справу з образами Docker. Додатки можуть створюватися за допомогою різних мов програмування й середовищ розробки: node, Java SE, Ruby, Python. Всі середовища працюють на Docker, але розроблювач навіть не знає, що він використовує контейнери.

Таким чином, залежно від того, що замовникові необхідно, він може вибрати кожної із трьох сценаріїв використання. Ми вважаємо, що Docker відмінно підходить для створення нових додатків. Але, крім того, не можна не відзначити потреба в контейнеризації наявних додатків. Всі продукти, які підтримує Oracle, можуть бути розгорнуті в контейнерному середовищі.

Кожний із замовників, з ким мені доводило зустрічатися, займається розробками програмного забезпечення тією чи іншою мірою, оскільки воно лежить у самому серці інновацій і дозволяє компанії виділитися серед конкурентів. Розроблювачів може бути небагато, але тим, що є, необхідно сфокусуватися на інноваціях.

У деяких галузях (роздрібна торгівля, фінанси, телекомунікації) потреба в диференціації вище, ніж в інших. Перераховані галузі перетерплюють серйозну трансформацію, і програмна розробка лежить у її основі. Але перетворення відбуваються всюди. Наприклад, державні інститути розробляють безліч додатків для надання кращих сервісів громадянам з використанням нових технологій і мобільних пристроїв.

Насправді тестування включене в процес розробки – у цьому суть сучасних підходів DevOps, що передбачають автоматизацію тестування й складання програм. Нові стандарти покликані спростити завдання забезпечення безпеки. Наприклад, технологія OAuth одержала широке поширення як спосіб забезпечити керування ідентифікацією в різних застосунках.

З переходом у хмару зростає потреба в консолідації ідентифікаційної інформації. Фрагментація ідентифікаторів породжує серйозну проблему для компаній: користувач може бути представлений в SaaS-застосунках у внутрикorporативних застосунках і в хмарних сервісах третіх сторін. Тому ми створили хмарний сервіс Identity Cloud Service, за допомогою якого ідентифікаційні дані можна консолідувати в хмарі. Крім того, Identity Cloud Service здійснює об'єднання (федерацію) ідентифікаторів з різних додатків і їхнє надання зовнішнім сервісам. Таким чином, досить один раз зареєструватися, щоб одержати доступ до всіх додатків при наявності відповідних прав.

Інший важливий момент, що стосується безпеки в хмарах, – ідея брокера безпеки хмарного доступу (Cloud Access Security Broker, CASB). Компанії найчастіше не знають, що саме роблять у хмарах їхні співробітники. CASB реєструється у всіх хмарах від вашого ім'я й надає детальні дані про використання хмарних сервісів. При виявленні якого-небудь аномального поведіння (наприклад, завантаження понад 20 Мбайт із корпоративної системи CRM) генеруються тривожні повідомлення. Популярність подібних сервісів росте, оскільки використання CASB стає обов'язковою вимогою в багатьох організаціях.

Можна ще довго говорити про те, що робить для реалізації мер захисту на всіх рівнях хмарної платформи. Підкреслю лише, що безпеки приділяється першорядна увага на всіх рівнях – від елементного до рівня керування.

Що таке хмари й коли має сенс будувати хмарні рішення? І якщо будувати, то які платформи використовувати? Чи потрібно надавати клієнтам хмарні сервіси? А може краще використовувати віртуалізацію? І чим відрізняється віртуалізація від хмар? Ці питання задають всі ІТ і «не ІТ» компанії: від великих операторів зв'язку до невеликих стартапів. Давайте спробуємо розібратися! Виникає питання – якщо й те й інше пов'язане з віртуалізацією і мережами, те це вийшли хмари? Адже всі ми прекрасно знаємо, що хмари – це віртуалізація ресурсів десь у мережі. Я спробував знайти формальне визначення хмар, але не знайшов. Визначення були розпливчасті й нечіткі, як справжні хмари. Для початку знайшов стандарт Cloud Computing, що розробляє NIST (National Institute of Standards and Technology). Він виділяє 3 виміру хмарних сервісів:

Хмари

Тобто чіткого визначення хмари немає навіть у стандарті. Розглянемо докладніше всі виміри:

Модель обслуговування або сервісна модель

Наступні поняття напевно всі чули:

– **IaaS (Infrastructure as a Service)** – постачальник надає тільки обчислювальні ресурси (servers, storage, processing, memory, network bandwidth). При цьому може надаватися набір підготовлених образів OS, які розвертаються за кілька хвилин. Також надається мережна зв'язність між віртуальними машинами й додаткові дискові ресурси. Особливою частиною IaaS є MaaS (Metal as a Service) – надання доступу до голого заліза без OS (bare-metal servers). Приклад постачальника IaaS: AWS. Платформи для побудови IaaS: VMware, OpenStack, Azure.

– **PaaS (Platform as a Service)** – постачальник надає операційну систему й певний софт для самостійної реалізації замовником своїх послуг, наприклад, DB, CMS. Приклади: OpenShift, Heroku, Google Cloud Platform, Cloud Foundry.

– **SaaS (Software as a Service)** – постачальник надає софт для використання замовником, наприклад: Microsoft Office 360, Microsoft Exchange і т.п.

Є думка, що справжні хмари це тільки PaaS і SaaS, але стандарт ця думка не підтверджує.

Модель розгортання

Тепер про моделі розгортання, які теж на слуху:

– **Private cloud (Частка хмара)** – інфраструктура розвертається усередині організації, тим самим забезпечується безпека й продуктивність.

– **Public cloud (Публічна хмара)** – інфраструктура призначена для надання послуг стороннім замовникам. Наприклад, AWS, Google Cloud. Переваги тут очевидні: не потрібний обслуговуючий персонал, швидкорозширяємі ресурси, платиш тільки за те, що спожив (години, гігабайти), можливість зосередитися на ідеї або бізнесі, а не на обслуговуванні інфраструктури. Добре підходять для швидкого розгортання невеликих систем, для організації стартапів.

– **Community cloud (Суспільна хмара)** – інфраструктура розділяється між декількома організаціями зі схожими концепціями споживання ресурсів, політиками в безпеці й т.п. Споживачів менше ніж у публічній хмарі, але більше чим у частці. Приклад: Google Apps for Government.

– **Hybrid cloud (Гібридна хмара)** – це комбінація із двох або більше різних хмарних інфраструктур (приватних, публічних або суспільних). Серед гібридних хмар є цікаві кейси використання, наприклад: усередині частки хмари організація зберігає всю необхідну актуальну інформацію, а резервні копії прозора відправляються в публічну хмару.

Функціональні характеристики

А от на функціональні характеристики хмар найчастіше не звертають увагу, тоді як для хмар це невід'ємні вимоги:

– **Broad Access Network** – можливість доступу до ресурсів з Інтернету за допомогою різноманітних пристроїв.

– **Rapid Elasticity** – еластичність ресурсів. При необхідності швидке виділення додаткових ресурсів, аж до нескінченності.

– **Measured Services** – автоматичний вимір послуг і виставлення рахунків для оплати.

– **On-Demand Self Services, Resource Pooling** – виділення обчислювальних ресурсів (resource pooling) за запитом від замовників (on demand).

Виходить, щоб Сервіс мав право називатися хмарним, потрібно вміти відповісти на запитання «Де він перебуває по всім трьох вимірах?».

Зворотна сторона хмар

Але при всіх перевагах хмар існує ряд моментів, які можуть мати негативні наслідки, особливо в публічних хмарах:

– Хмарою порівняно легко управляти, але відновлення софта впливає на тисячі користувачів, що потенційно може привести до серйозних аварій.

– У публічних хмарах більші конфігурації не завжди дешевше, треба вважати TCO.

Виникає кілька питань до безпеки в публічних хмарах:

– Як обробляються конфіденційні дані?

- Чи дотримані вимоги регулятора до персональним даних?
- Наскільки надійна схоронність даних (дублювання, бекапування)?
- Який захист від хакерських атак?

Це всі, звичайно, не привід відмовитися від використання хмар, а причина більш ретельно розібратися в плюсах і мінусах.

Віртуалізація

Тепер кілька слів про віртуалізацію, без якої не може бути хмар. У віртуалізації також є кілька вимірів, які умовно можна назвати «Тип» і «Спосіб віртуалізації». Я коротко розскажу про усі.

Типи віртуалізації

Віртуалізація серверів

Звичайно під віртуалізацією мається на увазі розміщення декількох віртуальних серверів у рамках одного фізичного. Але може ще бути об'єднання декількох фізичних серверів в один логічний для рішення певного завдання:

Розподіленість + віртуалізація = GRID системи

Віртуалізація ресурсів

Ресурси – це оперативна пам'ять, жорсткі диски, процесори. Вони також можуть бути нарізані й роздані вроздріб різним користувачам.

Віртуалізація додатків

Віртуалізація додатків – це те, що ми вже знаємо як PaaS і SaaS.

Способи віртуалізації

Повна віртуалізація й паравіртуалізація

Основні методи віртуалізації – це повна віртуалізація й паравіртуалізація. Схема обох методів дуже схожа. Є гіпервізор і віртуальні машини з гостьовими OS. При повній віртуалізації в гостьову OS не вносяться ніякі зміни. При паравіртуалізації встановлюються оптимізовані образи для певного гіпервізору. Це дозволяє максимально використовувати апаратні ресурси й не вимагає ніяких змін від додатків. Прикладом системи, що реалізує повну віртуалізацію, є VMware, приклад паравіртуалізації – Xen і KVM. В KVM особливістю є те, що гостьова OS може бути тільки Linux.

Ще виділяють кілька способів віртуалізації.

Віртуалізація рівня операційної системи

Особливістю є, що гостьова OS може бути тільки одна. Прикладом віртуалізації на рівні OS є Linux-VServer.

Контейнеризація

Останнім часом усе більше використовуються системи контейнеризації, такі як Docker або Kubernetes. Вони дозволяють автоматично розвертати підготовлені образи OS в основному для цілей автоматичного тестування й для систем CI. Контейнери дуже схожі на віртуальні машини, але для них не потрібно гіпервізор, а тільки відповідний движок.

Емуляція встаткування

При цьому способі віртуалізації VM повністю емулює роботу певного встаткування. З одного боку, це дає можливість, наприклад, на одному процесорі емулювати інший тип процесора. З іншого боку, зрозуміло, що при цьому буде з роботи в десятки разів. Приклад емулятора – це Vochs.

Емуляція бібліотек OS

І для повноти картини додам емуляцію бібліотек. Це спосіб, при якому емулюється не вся OS, а тільки частина. Наприклад, Wine в Linux – емуляція бібліотек для Windows-додатків.

Виходить, що ми можемо віртуалізувати різні системи різними методами. Тобто бачимо, що існуючими способами й типами віртуалізації можна вирішити безліч завдань. Виникає логічне запитання, коли ж має сенс розвертати хмарну платформу?

Хмарні платформи розташовуються над набором віртуальних машин, повністю ізолюють застосунок як від заліза, так і від структури віртуального середовища. Хмарні

платформи використовуються для автоматичного й ручного scale in / scale out, запуску / зупинки / конфігурування VM і додатків. Коли має сенс залишатися у віртуалізації, а коли залишатися в хмарі? Концепція наступна: коли всього багато – хмара, мало – віртуалізація:

- Для завдань із більшим числом віртуальних серверів – краще хмара, для завдань із 10-20 віртуалками можна не розвертати хмарну платформу.
- Якщо потрібно постійно міняти конфігурацію ресурсів – хмара.
- Багато користувачів, даних, клієнтів – хмара.

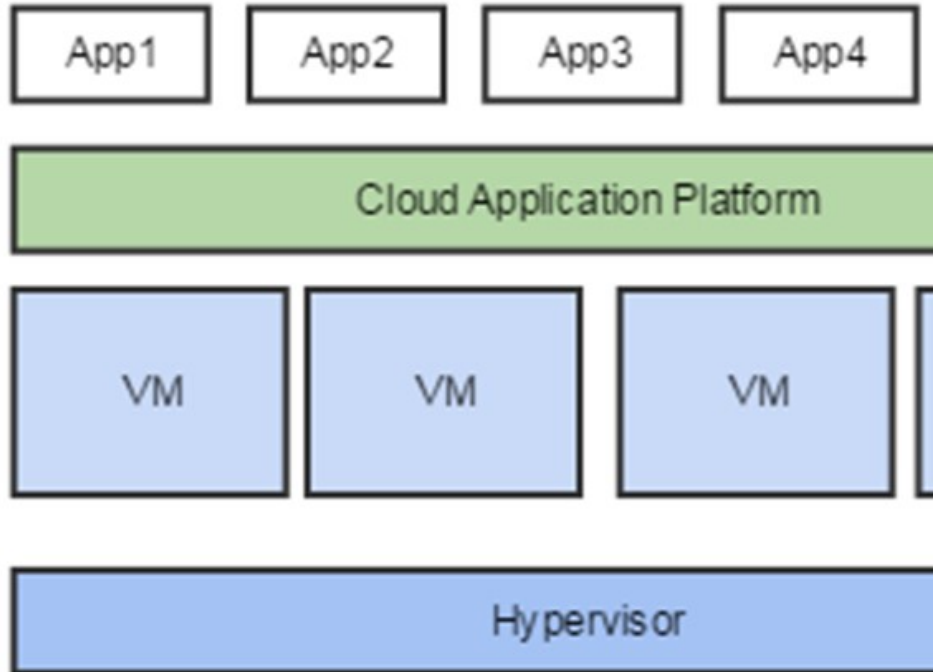


Рисунок 1 – Структурна схема системи

Кілька слів про платформи для побудови хмар. Інформації з них маса, тому приведу тільки суху вижимку.

Платформи для побудови хмар

Для побудови хмар зараз є дві основні платформи, інші (Eucalyptus, Cloudstack, Microsoft Azure) займають несуттєву частку.

- VMware – платформа vSphere. VMware зібрала величезну кількість технологій, веде розробку власної версії OpenStack.

- OpenStack – утворився в 2010 році, є комплексом проектів вільного програмного забезпечення, що може бути використаний для створення інфраструктурних хмарних сервісів і хмарних сховищ. Організація OpenStack Foundation (2012) нараховує більш ніж 1000 організацій і більше 30 000 учасників. Основні постачальники стека: Mirantis, RedHat, HP.

Хмарні платформи звичайно використовуються замовниками для реалізації наступних завдань:

- Робота із часто мінливою конфігурацією заліза, про що ми говорили вище.
- Надання інтернет-послуг і інші мережні додатки. Додатки повинні бути «Cloud Ready».
- Telco Cloud (вони ж SDN & NFV).
- Збір, зберігання, обробка BigData.

З отриманих даних можна зробити вивід, що для мобільних операторів зв'язку надання хмарних сервісів не є пріоритетним завданням, вони більше зосереджені на розвитку своїх мереж і сервісів. До операторів зв'язку потрібно звертатися за мережними послугами, такими як: віртуальна АТМ, віртуальна мережа, M2M и взагалі за всім тим, що може

надаватися на SDN & NFV технологіях: віртуальні Firewall, DPI і т.п. А за обчислювальними хмарами краще йти до спеціалізованих провайдерів.

Висновки. У статті наведені теоретичне узагальнення й рішення наукового завдання дослідження методів інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління. Рішення даного завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.
- Досліджена система інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.
- На основі отриманих результатів досліджень створена програмна реалізація системи інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління.

Розроблені алгоритми дозволяють успішно вирішувати завдання інтелектуальної інтеграції існуючих компонент ІТ-системи з хмарними службами наступного покоління. Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Список літератури

1. Al-Azzeh, J., Ayyoub, B., Mesleh, A., Smirnova, T., Gnatyuk, S., Driciev, O., Smirnov, O., Dorenskyi, O. «Cloud-Based Information System for Evaluating Caverns in the Process of Blasting Metal Surfaces of Details». *International Review on Modelling and Simulations* 18 (1), 2025. pp. 32-42.
2. Смірнова Т.В., Коноплицька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка*. 2024. №4(24), С. 6-27.
3. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». *Підводні технології*, 2024, № 13, с. 28-35.
4. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.
5. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.
6. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yanchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.
7. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.
8. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи*, 2023, том 7, № 2, С. 49-56.
9. Smirnova, T., Gnatyuk, S., Yudin, O., Sydorenko, V., Polozhentsev, A., «The Model for Calculating the Quantitative Criteria for Assessing the Security Level of Information and Telecommunication Systems». *CEUR Workshop Proceedings Volume 3156*, 2022, Pages 390-399.
10. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління*, № 2(70). 2022. С. 28-37.
11. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 3(69). С. 93-98.
12. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»*, № 2 (307). С. 46-52. 2022.
13. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості

- до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» Системи управління, навігації та зв'язку, 2022, № 1(67). С. 84-89.
14. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». *Communications in Computer and Information Science*, 2021, vol 1486. Springer, Cham. pp 169-184.
 15. Smirnov O., Kuznetsov A., Kiiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.
 16. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.
 17. Smirnov O., Kuznetsov A., Kiiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.
 18. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) *Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies*, vol 48. Springer, Cham. 2021. pp 557-587.
 19. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings Volume 2616*, 2020, Pages 125-136.
 20. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». *CEUR Workshop Proceedings Volume 2616*, 2020, Pages 366-379.
 21. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», *CEUR Workshop Proceedings Volume 2608*, 2020, Pages 633-645.
 22. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». *International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019*. P.22-28.
 23. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». *International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019*. – P. 393-407.
 24. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.
 25. Smirnov, O., Odarchenko, R., Abakumova, A., Usik, P., Kundyz, M., «QoE optimization technique for media delivery in 5G networks». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019. P.597-601.
 26. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». *CEUR Workshop Proceedings, Vol 2588*, P. 90-106, 2019.
 27. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.
 28. Smirnov, O., Kuznetsov, A., Kiiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 353-358.
 29. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 347-352.
 30. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», *CEUR Workshop Proceedings Volume 2353*, CEUR Workshop Proceedings 2019, Pages 618-629.
 31. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», *Telecommunications and Radio Engineering. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.*