План лекції.

Тема 3. Моніторинг віртуалізованих середовищ

- Вступ. Роль віртуалізації у сучасних IT-інфраструктурах
- Класифікація віртуалізованих середовищ
- Архітектура віртуалізованого середовища як об'єкта моніторингу
- Об'єкти моніторингу у віртуалізованих середовищах
- Ключові метрики та показники моніторингу
- Особливості та проблеми моніторингу віртуалізації
- Інструменти моніторингу віртуалізованих середовищ
- Підходи до побудови системи моніторингу
- Безпека та надійність моніторингу віртуалізованих середовищ
- Практичні приклади та типові сценарії

Вступ. Роль віртуалізації у сучасних IT-інфраструктурах

➢ *Еволюція обчислювальних середовищ: від фізичних серверів до віртуалізації*

Щоб зрозуміти, чому сьогодні ми так багато говоримо про віртуалізацію і чому її моніторинг став окремою, складною задачею, варто почати з невеликого історичного відступу.

На ранніх етапах розвитку корпоративних інформаційних систем обчислювальна інфраструктура будувалася максимально просто і прямолінійно. Один фізичний сервер — одна задача або один сервіс. Сервер встановлювався у серверній, на нього інсталювалася операційна система, прикладне програмне забезпечення, і він роками виконував одну й ту саму функцію. Такий підхід здавався логічним і безпечним: якщо сервіс критичний, він повинен мати власний «залізний» сервер і ні з чим не ділити ресурси.

Проте з часом почали проявлятися серйозні недоліки такої моделі. По-перше, більшість серверів були суттєво недовантажені. Процесор, пам'ять, дискова підсистема використовувалися лише на 10–20 %, але при цьому споживали електроенергію, займали місце, потребували охолодження та обслуговування. По-друге, масштабування інфраструктури було повільним і дорогим: щоб запустити новий сервіс, потрібно було закупити нове обладнання, чекати його постачання, налаштовувати та вводити в експлуатацію. По-третє, зростала складність адміністрування — десятки або сотні фізичних серверів потрібно було окремо моніторити, оновлювати та резервувати.

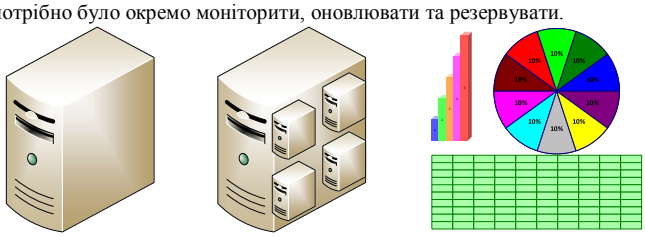


Рис.3.1. Фізичний хост та віртуалізація кількох хостів.

Саме на цьому етапі виникла ідея більш ефективного використання обчислювальних ресурсів. Інженери почали замислюватися: а що, якщо на одному фізичному сервері можна запускати декілька ізольованих обчислювальних середовищ, кожне зі своєю операційною системою та застосунками? Так з'являється концепція віртуалізації.

Віртуалізація дозволила абстрагувати фізичне обладнання від програмного забезпечення. Фізичний сервер перестав бути жорстко прив'язаним до конкретного сервісу. Натомість з'явився гіпервізор — спеціальний програмний рівень, який керує ресурсами фізичного хоста і розподіляє їх між віртуальними машинами. Кожна віртуальна машина сприймає себе як повноцінний сервер, хоча насправді ділить процесор, пам'ять, диски та мережу з іншими VM.

Цей перехід став справжньою революцією для IT-інфраструктур. Віртуалізація зробила можливим швидке розгортання нових сервісів — тепер сервер можна було створити за кілька хвилин, а не тижнів. Зросла гнучкість: віртуальні машини стало легко переносити між фізичними хостами, резервувати, клонувати та масштабувати. Значно покращилася економічна ефективність використання ресурсів.

Однак разом із цими перевагами змінився і сам характер обчислювального середовища. Інфраструктура перестала бути статичною. Віртуальні машини могли з'являтися, зникати, мігрувати між серверами, змінювати споживання ресурсів у динамічному режимі. З точки зору адміністратора або інженера з експлуатації, система стала значно менш прозорою. Якщо раніше було зрозуміло, що проблема продуктивності пов'язана з конкретним фізичним сервером, то у віртуалізованому середовищі причина може ховатися на будь-якому рівні: у гіпервізорі, у конфігурації ресурсів, у взаємному впливі віртуальних машин одна на одну.

Саме тут моніторинг починає відігравати принципово нову роль. Він перестає бути просто інструментом спостереження за завантаженням процесора чи пам'яті. У віртуалізованих середовищах моніторинг стає засобом відновлення прозорості системи, способом побачити реальну картину того, як фізичні ресурси перетворюються на логічні обчислювальні середовища і як ці середовища впливають одне на одного.

Таким чином, віртуалізація не просто змінила технічну реалізацію серверів. Вона докорінно змінила підхід до побудови, експлуатації та моніторингу IT-інфраструктур, заклавши основу для наступних етапів розвитку — високодоступних кластерів, хмарних платформ і контейнеризованих середовищ, про які ми будемо говорити далі в курсі.

➢ *Чому традиційний моніторинг фізичних систем є недостатнім*

Коли ми говоримо про традиційний моніторинг, зазвичай маємо на увазі підхід, який формувався в епоху фізичних серверів. Тоді все було відносно просто і прозоро. Існував конкретний сервер, з чітко визначеним апаратним забезпеченням, встановленою операційною системою та набором сервісів. Моніторинг зводився до спостереження за базовими показниками: завантаженням процесора, використанням оперативної пам'яті, станом дисків, мережевою активністю. Якщо сервер «гальмує», значить, один із цих ресурсів перевантажений — і це легко простежити.

У такій моделі між фізичним обладнанням і сервісом існував прямий, майже лінійний зв'язок. Один сервер — один набір метрик — одна точка відповідальності. Традиційні системи моніторингу прекрасно справлялися з цим завданням, адже вони були створені саме для таких статичних і передбачуваних середовищ.

Проте з появою віртуалізації цей зв'язок почав руйнуватися. Фізичний сервер більше не був «одиницею сервісу». Він перетворився на платформу, всередині якої одночасно працюють десятки віртуальних машин, кожна зі своїми потребами, навантаженням і життєвим циклом. І традиційний моніторинг, який бачить лише фізичний рівень, виявився сліпим до того, що відбувається всередині.

Уявімо типову ситуацію: моніторинг фізичного сервера показує, що CPU завантажений лише на 40 %, пам'яті ще достатньо, дискова підсистема працює в межах норми. З точки зору класичного підходу — все гаразд. Але при цьому користувачі скаржаться на повільну роботу конкретного сервісу, який запущений у віртуальній машині. Традиційний моніторинг не здатен пояснити цю проблему, адже він не бачить черг процесора на рівні гіпервізора, не враховує конкуренцію між VM, не знає про затримки, пов'язані з плануванням ресурсів.

Ще одна фундаментальна проблема полягає в тому, що у віртуалізованому середовищі ресурси перестають бути «гарантованими» у фізичному сенсі. Процесорні ядра, оперативна пам'ять, диски — усе це може бути перевиділено (overcommit). З погляду віртуальної машини ресурс нібито існує, але фактично він ділиться з іншими VM. Традиційний моніторинг фізичної системи не бачить цього рівня абстракції і, відповідно, не може коректно інтерпретувати реальне навантаження.

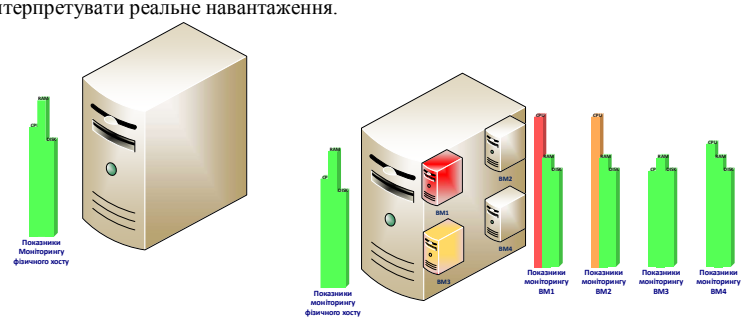


Рис.3.2. Фізичні метрики більше не відображають реальні статуси сервісів.

Крім того, змінюється сама динаміка інфраструктури. У фізичному світі сервери рідко «мігрують». Якщо система встановлена на конкретному обладнанні, вона там і залишається. У віртуалізованому середовищі віртуальна машина може автоматично переміститися з одного фізичного хоста на інший, наприклад, для балансування навантаження або під час обслуговування. Для традиційного моніторингу це виглядає як зникнення одного об'єкта і поява іншого, без розуміння, що це той самий сервіс, який просто змінив фізичне місце існування.

Також варто врахувати, що класичний моніторинг зазвичай орієнтований на окремі вузли, а не на взаємозв'язки між ними. У віртуалізованому середовищі проблеми часто виникають саме на стику рівнів: між віртуальною машиною і гіпервізором, між кількома VM, що конкурують за ресурси, або між віртуальними мережами та фізичною інфраструктурою. Традиційні підходи просто не мають інструментів для аналізу таких залежностей.

У результаті адміністратор отримує парадоксальну ситуацію: система моніторингу показує «зелений» статус фізичних серверів, але реальна якість роботи сервісів погіршується. Це підриває довіру до моніторингу як такого і змушує шукати причини проблем уже постфактум, у ручному режимі.

Саме тому перехід до віртуалізованих середовищ вимагає переосмислення підходів до моніторингу. Недостатньо просто спостерігати за фізичними метриками — необхідно бачити всю багаторівневу картину: від апаратного забезпечення до гіпервізора і конкретної віртуальної машини. Лише тоді моніторинг знову стає інструментом не лише фіксації проблем, а й їхнього попередження.

- ***Місце моніторингу віртуалізованих середовищ у загальній системі IT Operations, ITSM та SRE***

Певно є сенс нагадати термінологію/абревіатури, що використовуються далі.

- ***IT Operations (IT Ops)*** – повсякденна експлуатація IT-систем. Фахівці з IT Operations отримують типові ролі System Administrator, Network Engneer, Operations Engineer. Фокус їх діяльності зосереджений на стабільності та безперервності роботи. Розгорнуто це звучить:
  - підтримка серверів, мереж, БД, хмари
  - слідкують, щоб сервіси працювали 24/7
  - реагування на інциденти (щось упало → підняти)
  - роблять бекапи, оновлення, патчі
  - керують доступами та безпекою
- ***ITSM (IT Service Management)*** *– процеси та правила, як надавати IT-послуги бізнесу. ITSM не команда, а фреймворк управління фокус* діяльності якого зосереджено на процесах, стандартах, прозорості та контролі. Включає:
  - Incident Management — аварії
  - Problem Management — пошук кореня проблем
  - Change Management — контроль змін
  - Service Request — запити користувачів
  - SLA / SLO — рівні сервісу
- ***SRE (Site Reliability Engineering)*** – інженерний підхід до надійності, який прийшов з Google. Друге значення абревіатури SRE = Software Engineering + Operations. Фокус діяльності SRE зосереджено на надійності (саме через код) та автоматизації. Фахівці SRE:
  - автоматизують Ops (Infrastructure as Code)
  - створюють системи моніторингу й алертингу
  - визначають SLO / SLI / Error Budget
  - працюють з масштабуванням і продуктивністю
  - мінімізують ручну роботу й on-call біль

	IT Ops	ITSM	SRE
Що це	Команда/функція	Процеси	Інженерна роль
Підхід	Реактивний	Регламентований	Проактивний
Основний фокус	Щоб не впало	Щоб правильно працювали сервіси	Щоб не падало і масштабувалось
Автоматизація	Часткова	Мінімальна	Максимальна

Як це працює разом

- IT Ops — “гасить пожежі”
- ITSM — “визначає правила гри”
- SRE — “робить так, щоб пожеж було менше”

Коли віртуалізація стає основою інфраструктури, змінюється не лише технічна архітектура, а й підхід до експлуатації IT-систем загалом. Моніторинг у цьому контексті вже не можна розглядати як ізольований інструмент для системних адміністраторів. Він стає центральним елементом всієї операційної діяльності — тим джерелом даних, на якому базуються управлінські, технічні та організаційні рішення.

У класичному розумінні IT Operations відповідали за стабільну роботу інфраструктури: сервери повинні бути доступні, мережі — функціонувати, сервіси — відповідати заданому рівню якості. У фізичному світі це означало реагування на збої конкретних серверів або пристроїв. У віртуалізованому середовищі сама інфраструктура стає більш абстрактною, а отже й операційна діяльність змінює свій фокус. IT Operations більше не керують окремими серверами — вони керують середовищем, у якому постійно змінюється склад об’єктів.

Саме моніторинг у цьому випадку виконує роль «органів чуття» операційної команди. Він дозволяє побачити, як віртуальні машини розподіляються між хостами, як змінюється навантаження, де виникають точки ризику. Без якісного моніторингу IT Operations втрачають контроль над середовищем, яке є динамічним за своєю природою.

Якщо ж подивитися на це питання з точки зору ITSM, тобто управління IT-сервісами, роль моніторингу стає ще більш фундаментальною. ITSM фокусується не на серверах і не на віртуальних машинах, а на сервісах, які отримує бізнес або кінцевий користувач. Віртуалізація тут відіграє роль прихованого шару, який не цікавить користувача напряму, але критично впливає на якість сервісу.

Моніторинг віртуалізованих середовищ у рамках ITSM виконує роль мосту між технічною інфраструктурою і сервісним рівнем. Саме завдяки йому можна зрозуміти, чому порушується SLA, які технічні події призводять до інцидентів, і де саме в інфраструктурі виникає першопричина проблеми. Без цього ITSM-процеси — управління інцидентами, проблемами, змінами — перетворюються на реактивні та малоефективні, адже їм бракує достовірних даних.

Окрему увагу варто приділити сучасному підходу SRE, який поєднує інженерне мислення з експлуатацією систем. У SRE моніторинг розглядається не просто як спосіб «дізнатися, що щось зламалося», а як інструмент вимірювання надійності системи. Саме тут моніторинг віртуалізованих середовищ виходить на новий рівень значущості.

Для SRE важливо розуміти, як поведінка віртуальної інфраструктури впливає на досягнення SLO та дотримання SLI. Наприклад, нестача ресурсів на рівні гіпервізора або агресивний overcommit можуть формально не виглядати як аварія, але поступово знижувати якість сервісу. Без глибокого моніторингу ці процеси залишаються непомітними до того моменту, коли порушується надійність системи в цілому.

Таким чином, у сучасних IT-організаціях моніторинг віртуалізованих середовищ перестає бути допоміжною функцією. Він стає спільною точкою перетину IT Operations, ITSM та SRE. Для операційних команд — це інструмент контролю та стабільності. Для сервісного управління — джерело об’єктивних даних про якість сервісів. Для SRE — основа для інженерного підходу до надійності та масштабування.

І саме в цьому полягає ключова ідея: без системного моніторингу віртуалізованих середовищ жодна з цих моделей — ні класична експлуатація, ні ITSM, ні SRE — не може повноцінно працювати. Віртуалізація робить інфраструктуру гнучкою, але лише моніторинг робить її керованою.

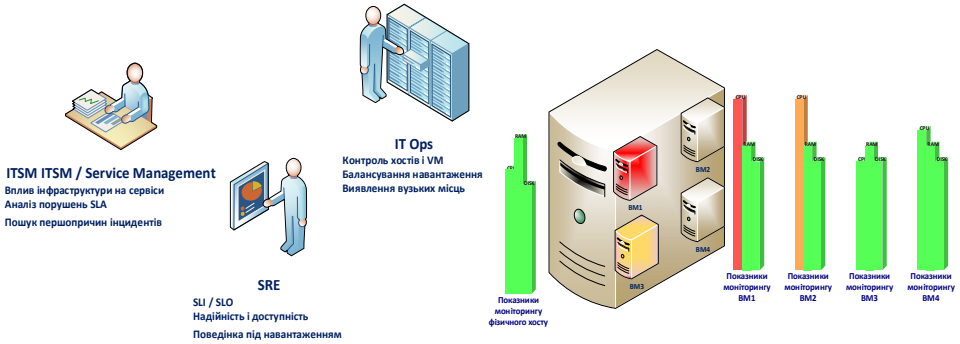


Рис.3.3. Моніторинг — єдине джерело даних для операцій, сервісів і надійності.

➢ ***Зв’язок теми з попереднім модулем: критичні та нетипові об’єкти IT-інфраструктури***

Перш ніж перейти безпосередньо до технічних аспектів моніторингу віртуалізованих середовищ, важливо усвідомити, що ця тема не з’являється в курсі випадково і не є відірваною від попереднього матеріалу. Навпаки, вона логічно продовжує і розвиває ідеї, закладені в першому модулі, присвяченому моніторингу критичних та нетипових об’єктів IT-інфраструктури.

У першому модулі ми розглядали інфраструктурні компоненти, від яких безпосередньо залежить стабільність бізнес-процесів і безпека організації. Ми говорили про критичні об’єкти — ті, відмова або деградація яких має непропорційно великий вплив на всю систему. Водночас ми звертали увагу на нетипові об’єкти, які складно моніторити через їхню специфіку, нестандартні протоколи, обмежені ресурси або відсутність класичних засобів спостереження.

Віртуалізовані середовища, на перший погляд, можуть не виглядати ні критичними, ні нетиповими. Проте на практиці вони поєднують у собі риси обох категорій. З одного боку, сучасна віртуалізована платформа часто є фундаментом усієї IT-інфраструктури. На ній працюють бази даних, бізнес-застосунки, сервіси аутентифікації, системи резервного копіювання. Відмова або некоректна робота гіпервізора чи кластера віртуалізації миттєво впливає на десятки або сотні сервісів одночасно. У цьому сенсі віртуалізоване середовище є класичним прикладом критичного об’єкта, навіть якщо кожна окрема віртуальна машина такою не здається.

З іншого боку, віртуалізовані середовища мають низку ознак, які роблять їх нетиповими з точки зору моніторингу. Ми вже бачили подібні виклики у випадку IoT, SCADA або периферійних систем: відсутність прямого доступу до ресурсів, багаторівнева архітектура, складні залежності між компонентами. У віртуалізації ці проблеми проявляються у вигляді додаткового шару абстракції — гіпервізора, який приховує реальну взаємодію між фізичним обладнанням і сервісами.

Таким чином, багато підходів, які ми розглядали в першому модулі, безпосередньо застосовуються і тут. Це і питання визначення критичності, і вибір релевантних метрик, і необхідність адаптації засобів моніторингу до специфіки об’єкта. Як і у випадку нетипових об’єктів, у віртуалізованому середовищі стандартні інструменти часто дають неповну або оманливу картину, якщо їх використовувати без урахування архітектурних особливостей.

Більше того, віртуалізація часто стає точкою концентрації ризиків. Один фізичний збій, одна помилка конфігурації або один некоректний сценарій навантаження можуть вплинути одразу на велику кількість сервісів. Це підсилює вимоги до моніторингу, роблячи його не просто інструментом спостереження, а механізмом раннього попередження і управління ризиками — саме так, як ми розглядали у контексті критичних об'єктів.

Отже, перехід до теми моніторингу віртуалізованих середовищ є логічним продовженням першого модуля курсу. Якщо раніше ми вчилися працювати з об'єктами підвищеної відповідальності та нестандартної природи, то тепер застосовуємо ці самі принципи до сучасної основи більшості IT-інфраструктур. Це створює місток до наступних тем — високодоступних кластерів і контейнеризованих середовищ, де складність, динаміка і вимоги до моніторингу зростають ще більше.

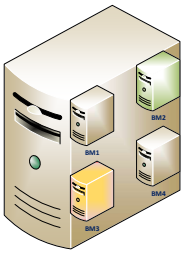
Класифікація віртуалізованих середовищ

➢ *Класифікація віртуалізованих середовищ за рівнем віртуалізації*

Коли ми говоримо про віртуалізовані середовища, дуже важливо одразу визначитися, що саме ми маємо на увазі під словом «віртуалізація». У практиці експлуатації та моніторингу це поняття охоплює не одну технологію, а цілий спектр підходів, які відрізняються тим, **на якому рівні системи створюється абстракція** між фізичними ресурсами та програмним забезпеченням.

Саме тому доцільно почати з класифікації віртуалізованих середовищ за рівнем віртуалізації. Це дозволяє одразу зрозуміти, де саме виникають нові об'єкти моніторингу, які метрики стають релевантними і чому універсального підходу до спостереження за такими середовищами не існує.

❖ *Апаратна віртуалізація (hardware virtualization)*



Гіпервізор+ВМ

Найбільш класичним і водночас найпоширенішим видом є апаратна віртуалізація. Саме її зазвичай мають на увазі, коли говорять про віртуальні машини у корпоративних дата-центрах або хмарних середовищах.

У цьому випадку віртуалізація реалізується на рівні апаратного забезпечення за допомогою гіпервізора. Фізичний сервер з його процесором, пам'яттю, дисками та мережевими інтерфейсами перетворюється на пул ресурсів, з якого гіпервізор «збирає» віртуальні машини. Кожна з них отримує власну віртуальну апаратну конфігурацію і працює під управлінням повноцінної операційної системи.

З точки зору моніторингу, апаратна віртуалізація створює багаторівневу модель спостереження. Тепер недостатньо просто дивитися на фізичний сервер — необхідно одночасно контролювати стан хоста, гіпервізора і кожної окремої віртуальної машини. При цьому проблеми можуть виникати не на одному конкретному рівні, а внаслідок їхньої взаємодії. Наприклад, фізичний сервер може виглядати стабільним, але через особливості планування ресурсів окремі VM відчуватимуть дефіцит продуктивності.

Саме апаратна віртуалізація першою поставила питання про спеціалізований моніторинг, який здатен «бачити» внутрішню логіку розподілу ресурсів і пояснювати поведінку віртуальних машин.

❖ *Віртуалізація на рівні операційної системи (OS-level virtualization)*


Контейнери

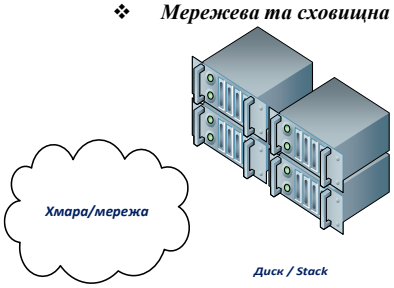
Наступний рівень — віртуалізація на рівні операційної системи. На відміну від апаратної віртуалізації, тут немає окремого гіпервізора, який емулює апаратне середовище. Натомість одна операційна система запускає кілька ізольованих середовищ, які спільно використовують її ядро.

Цей підхід довгий час залишався менш помітним, але з появою контейнерних технологій він став надзвичайно популярним. Контейнери можна розглядати як логічне продовження і розвиток ідей OS-level virtualization. Вони забезпечують легку, швидку ізоляцію процесів і ресурсів без необхідності запускати повноцінну операційну систему для кожного середовища.

З точки зору моніторингу, віртуалізація на рівні ОС створює принципово інші виклики. Тут межа між «хостом» і «віртуальним середовищем» стає значно тоншою. Традиційні метрики операційної системи більше не можна інтерпретувати однозначно, адже один і той самий процесор або обсяг пам'яті використовується одразу багатьма ізольованими середовищами.

Крім того, життєвий цикл таких середовищ значно коротший і динамічніший. Вони можуть створюватися і знищуватися за секунди, що вимагає від систем моніторингу зовсім іншого підходу до збору та агрегації даних. Саме тому віртуалізація на рівні ОС часто розглядається як проміжний етап між класичною віртуалізацією і контейнеризованими платформами, які ми будемо детально розглядати пізніше.

❖ *Мережева та сховищна віртуалізація як складові середовища*



Окрім обчислювальної віртуалізації, сучасні віртуалізовані середовища майже завжди включають мережеву та сховищну віртуалізацію. Хоча вони рідко розглядаються окремо в рамках базових курсів, для моніторингу їхня роль є критично важливою.

Мережева віртуалізація дозволяє створювати логічні мережі, комутатори, маршрутизатори і міжмережеві екрани, які існують незалежно від фізичної топології. Для моніторингу це означає появу додаткового рівня, на якому можуть виникати затримки, втрати пакетів або помилки конфігурації, невидимі на фізичному рівні.

Сховищна віртуалізація, у свою чергу, приховує реальну структуру дискових ресурсів, представляючи їх у вигляді логічних томів або datastore. Віртуальна машина може «бачити» диск, але фактичний шлях даних до фізичного носія може бути складним і багаторівневим. Це робить аналіз продуктивності введення-виведення одним із найскладніших аспектів моніторингу віртуалізованих середовищ.

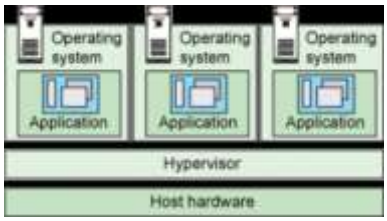
Таким чином, сучасне віртуалізоване середовище — це не просто сукупність віртуальних машин. Це багаторівнева система абстракцій, де обчислювальні, мережеві та сховищні компоненти тісно переплетені між собою. І розуміння рівнів віртуалізації є першим, необхідним кроком до побудови ефективної системи моніторингу.

➢ *Класифікація віртуалізованих середовищ за типом гіпервізора*

Після того як ми розібралися з рівнями віртуалізації, логічно зробити наступний крок і подивитися на те, як саме реалізується керування віртуальними машинами. Ключову роль тут відіграє гіпервізор — компонент, який фактично визначає архітектуру середовища, його надійність, продуктивність і, що для нас особливо важливо, підхід до моніторингу.

Саме за типом гіпервізора віртуалізовані середовища традиційно поділяють на два великі класи: гіпервізори типу 1 і типу 2. Цей поділ здається формальним лише на перший погляд. Насправді він має глибокі практичні наслідки для експлуатації та спостереження за системою.

❖ ***Гіпервізори типу 1 (bare-metal)***



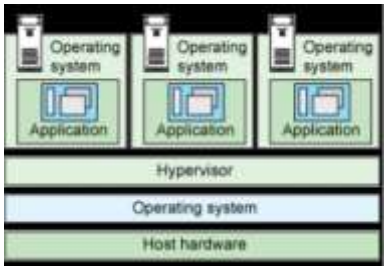
Гіпервізори типу 1 працюють безпосередньо на апаратному забезпеченні, без проміжного шару у вигляді повноцінної хостової операційної системи. Саме тому їх часто називають bare-metal гіпервізорами. У такій архітектурі фізичний сервер завантажується одразу в гіпервізор, який бере на себе керування всіма апаратними ресурсами і розподіляє їх між віртуальними машинами.

Типовими представниками цього класу є VMware ESXi, Microsoft Hyper-V Server та KVM у складі серверних Linux-систем. Саме ці платформи найчастіше використовуються у корпоративних дата-центрах і хмарних інфраструктурах, де на перший план виходять продуктивність, масштабованість і надійність.

З точки зору моніторингу, гіпервізори типу 1 формують відносно чітку ієрархію об'єктів. Є фізичний хост, є гіпервізор як окремий керуючий рівень, і є віртуальні машини. Така структура дозволяє будувати багаторівневий моніторинг, у якому можна відстежувати як загальний стан хоста, так і поведінку кожної VM окремо.

Водночас саме у bare-metal середовищах найгостріше проявляються проблеми конкуренції за ресурси. Гіпервізор виконує складну роботу з планування процесорного часу, пам'яті та введення-виведення, і традиційні метрики операційних систем тут уже не дають повної картини. Моніторинг повинен враховувати специфічні показники, характерні саме для гіпервізора, і використовувати його внутрішні механізми збору даних та API.

❖ ***Гіпервізори типу 2 (hosted)***



Гіпервізори типу 2 працюють інакше. Вони запускаються як звичайний застосунок поверх повноцінної хостової операційної системи. Фактично віртуалізація в цьому випадку стає одним із сервісів, які надає хостова ОС.

До цього класу належать такі рішення, як VirtualBox або VMware Workstation. Їх найчастіше використовують у навчальних цілях, для тестування, розробки або демонстраційних середовищ. Вони простіші у встановленні та не вимагають виділеного сервера, але мають обмеження щодо продуктивності та масштабованості.

З погляду моніторингу, гіпервізори типу 2 створюють менш чітку межу між рівнями системи. Гіпервізор, віртуальні машини і хостова операційна система тісно переплетені між собою. Навантаження від віртуальних машин сприймається хостовою ОС як звичайні процеси, а проблеми на рівні хоста безпосередньо впливають на всі VM.

Це означає, що моніторинг у таких середовищах часто зводиться до спостереження за хостовою операційною системою, доповненого окремими метриками віртуалізації. З одного боку, це спрощує підхід, з іншого — обмежує глибину аналізу і ускладнює точне визначення причин проблем.

❖ ***Особливості моніторингу кожного типу гіпервізора***

Порівнюючи ці два підходи, можна сказати, що тип гіпервізора безпосередньо визначає складність і можливості моніторингу. У середовищах з гіпервізорами типу 1 моніторинг стає невід'ємною частиною інфраструктури і вимагає спеціалізованих інструментів, які «розуміють» внутрішню логіку гіпервізора. Саме тут з'являються такі поняття, як специфічні метрики планування ресурсів, події міграції віртуальних машин, стан кластерів віртуалізації.

У випадку гіпервізорів типу 2 моніторинг залишається ближчим до класичних підходів, але водночас втрачає частину глибини. Він більше орієнтований на спостереження за хостовою системою, ніж на аналіз внутрішніх процесів віртуалізації.

Таким чином, розуміння типу гіпервізора є критично важливим для побудови адекватної системи моніторингу. Воно дозволяє ще на етапі проєктування визначити, які метрики будуть ключовими, які інструменти доцільно використовувати і яких обмежень варто очікувати. Саме з цього усвідомлення починається перехід від формального спостереження до справжнього контролю за віртуалізованим середовищем.

➢ ***Класифікація віртуалізованих середовищ за моделлю розгортання***

Після того як ми розглянули рівні віртуалізації та типи гіпервізорів, варто зробити ще один важливий крок і подивитися на віртуалізовані середовища з практичної точки зору — де і в яких умовах вони розгортаються. Саме модель розгортання визначає не лише технічну архітектуру, а й підходи до управління, відповідальність сторін та можливості моніторингу.

На цьому рівні віртуалізація перестає бути лише технологією і стає частиною організаційної та бізнес-моделі IT-інфраструктури.

❖ ***On-premise віртуалізація***

On-premise віртуалізація — це класичний сценарій, у якому віртуалізоване середовище повністю розгортається і експлуатується всередині власної інфраструктури організації. Сервери, мережеве обладнання, системи зберігання даних і програмне забезпечення перебувають під повним контролем внутрішньої IT-команди.

У такій моделі організація самостійно відповідає за все: від фізичного стану обладнання до налаштувань гіпервізорів, політик резервування та моніторингу. З одного боку, це забезпечує максимальну прозорість і контроль. Інженери можуть бачити всі рівні системи — від температури процесора до затримок введення-виведення конкретної віртуальної машини.

З іншого боку, саме on-premise середовища накладають найвищі вимоги до моніторингу. Тут немає зовнішнього провайдера, на якого можна перекласти частину відповідальності. Якщо виникає проблема, саме внутрішня команда повинна мати інструменти і дані, щоб швидко визначити її причину. Тому моніторинг у таких середовищах зазвичай є максимально глибоким і охоплює всі рівні віртуалізації.

❖ ***Приватні та публічні хмари як форма віртуалізованого середовища***

З розвитком хмарних технологій віртуалізація вийшла за межі власних дата-центрів. У моделі IaaS — Infrastructure as a Service — віртуалізація стає послугою, яку надає хмарний провайдер. Фізична інфраструктура і гіпервізори перебувають у зоні відповідальності провайдера, а користувач працює вже з готовими віртуальними ресурсами.

У публічних хмарах цей розподіл відповідальності особливо помітний. Користувач має доступ до віртуальних машин, мереж і сховищ, але не бачить фізичних серверів і не контролює гіпервізор напряму. Це кардинально змінює підхід до моніторингу. Частина даних стає недоступною, а частина — надається у вигляді абстрактних метрик через API хмарної платформи.

Приватні хмари займають проміжне положення. Формально вони також будуються на основі віртуалізації і принципів хмарних обчислень, але розгортаються в межах однієї організації або виділеного середовища. Тут моніторинг може бути більш глибоким, ніж у публічній хмарі, але водночас повинен інтегруватися з хмарними механізмами керування і автоматизації.

В обох випадках — і в приватних, і в публічних хмарах — моніторинг зміщується від фізичних метрик до логічних показників: доступності, продуктивності, використання ресурсів і відповідності SLA. Це вимагає нових інструментів і нового мислення, у якому віртуалізація сприймається як сервіс, а не як набір серверів.

- ***Гібридні віртуалізовані середовища***

Окрему увагу заслуговують гібридні середовища, які сьогодні фактично стали стандартом для багатьох організацій. У такій моделі частина віртуалізованої інфраструктури залишається on-premise, а частина — переноситься у хмару. Причини можуть бути різними: вимоги безпеки, регуляторні обмеження, оптимізація витрат або потреба у швидкому масштабуванні.

Для моніторингу гібридні середовища є найбільш складними. Тут поєднуються різні технології, різні моделі відповідальності і різні джерела даних. Віртуальні машини можуть мігрувати між середовищами, сервіси — залежати від компонентів, розміщених у різних локаціях, а проблеми — виникати на стику цих світів.

У таких умовах моніторинг повинен виконувати не лише технічну, а й інтеграційну функцію. Він має об'єднувати дані з локальної інфраструктури і хмарних платформ, забезпечуючи єдину картину стану системи. Без цього гібридна віртуалізація швидко перетворюється на джерело неконтрольованих ризиків.

Таким чином, класифікація віртуалізованих середовищ за моделлю розгортання показує, що віртуалізація — це не лише питання технологій, а й питання контексту. Те, де і як розгортається середовище, безпосередньо впливає на глибину, доступність і цілі моніторингу. Саме це розуміння дозволяє перейти від абстрактної класифікації до практичного аналізу віртуалізованого середовища як об'єкта моніторингу.

Рис.3.4. Моделі розгортання віртуалізованих середовищ та межі моніторингу.

Архітектура віртуалізованого середовища як об'єкта моніторингу

Коли ми говоримо про моніторинг віртуалізованих середовищ, надзвичайно важливо чітко усвідомлювати, **що саме ми моніторимо**. Віртуалізоване середовище — це не окремий сервер і не набір віртуальних машин. Це багаторівнева система, у якій кожен рівень має власні характеристики, обмеження та типові проблеми. І головне — усі ці рівні тісно пов'язані між собою.

Саме тому віртуалізоване середовище доцільно розглядати як єдиний об'єкт моніторингу, але з чітким поділом на архітектурні рівні. Такий підхід дозволяє не лише фіксувати симптоми, а й знаходити першопричини проблем.

- ***Фізичний рівень: апаратне забезпечення хоста***

Найнижчим, але водночас фундаментальним рівнем є фізичний рівень. Це реальне апаратне забезпечення: процесори, оперативна пам'ять, дискові підсистеми, мережеві адаптери, блоки живлення та система охолодження. Саме тут закладені фізичні межі всієї віртуалізації.

З точки зору моніторингу, цей рівень часто здається знайомим і навіть «простим», адже багато підходів перейшли сюди з епохи фізичних серверів. Однак у віртуалізованому середовищі фізичний рівень набуває особливого значення. Один апаратний збій або деградація продуктивності може вплинути одразу на десятки віртуальних машин, що робить цей рівень критичним об'єктом спостереження.

Важливо розуміти, що моніторинг фізичного рівня у віртуалізованих середовищах — це не лише про «чи працює сервер». Це про те, чи здатен він стабільно підтримувати навантаження, яке на нього покладено гіпервізором і віртуальними машинами.

- ***Рівень гіпервізора***

Над фізичним рівнем розташовується гіпервізор — ключовий елемент усієї архітектури. Саме він є «диспетчером» ресурсів, який приймає рішення про те, яка віртуальна машина і коли отримує доступ до процесора, пам'яті, дисків і мережі.

З точки зору моніторингу, рівень гіпервізора є особливо важливим, але водночас складним. Це той рівень, на якому відбувається більшість процесів, невидимих для операційних систем віртуальних машин. Планування процесорного часу, керування пам'яттю, обробка операцій введення-виведення — усе це реалізується саме тут.

Без моніторингу гіпервізора віртуалізоване середовище стає «чорною скринькою». Зовні ми бачимо лише симптоми — уповільнення сервісів, зростання затримок, нестабільну роботу VM, — але не розуміємо, що саме відбувається всередині. Тому повноцінний моніторинг віртуалізації завжди починається з цього рівня.

- ***Рівень віртуальних машин***

Наступний рівень — це рівень віртуальних машин. Саме його найчастіше сприймають як основний об'єкт моніторингу, адже саме тут працюють прикладні сервіси, з якими взаємодіють користувачі.

З точки зору віртуальної машини, усе виглядає досить знайомо: є операційна система, процеси, пам'ять, файлові системи, мережеві інтерфейси. Проте у віртуалізованому середовищі ці метрики не можна інтерпретувати ізольовано. Навантаження всередині VM є лише відображенням того, як гіпервізор виділяє їй ресурси.

Саме тут часто виникає пастка: моніторинг показує, що у VM достатньо CPU або пам'яті, але сервіс працює повільно. Без кореляції з рівнем гіпервізора і фізичного хоста така ситуація залишається незрозумілою. Тому рівень віртуальних машин повинен розглядатися не як самодостатній, а як частина загальної архітектури моніторингу.

- ***Віртуальні мережі та сховища***

Окремого розгляду заслуговують віртуальні мережі та сховища, які пронизують усі попередні рівні. У віртуалізованому середовищі мережеві з'єднання і дискові ресурси зазвичай не мають прямого відповідника у фізичній інфраструктурі. Вони реалізуються через логічні комутатори, віртуальні інтерфейси, datastore та інші абстракції.

З точки зору моніторингу, саме ці компоненти часто стають джерелом найскладніших проблем. Затримки в мережі або введенні-виведенні можуть виникати на будь-якому з рівнів — від фізичного адаптера до логічного шару віртуалізації. Без комплексного підходу такі проблеми надзвичайно складно діагностувати.

➢ ***Взаємозв'язки між рівнями та їх вплив на моніторинг***

Ключова ідея цього розділу полягає в тому, що жоден із рівнів не існує окремо. Усі вони взаємодіють між собою, і саме ці взаємозв'язки формують реальну поведінку віртуалізованого середовища.

Проблема на фізичному рівні може проявитися як деградація продуктивності віртуальних машин. Некоректне планування ресурсів на рівні гіпервізора може виглядати як проблема всередині операційної системи VM. А перевантаження віртуальної мережі може бути помилково сприйняте як збій прикладного сервісу.

Тому ефективний моніторинг віртуалізованих середовищ завжди будується на принципі кореляції даних між рівнями. Він дозволяє не лише фіксувати події, а й розуміти причинно-наслідкові зв'язки між ними. Саме це відрізняє зрілу систему моніторингу від простого набору графіків і алертів.

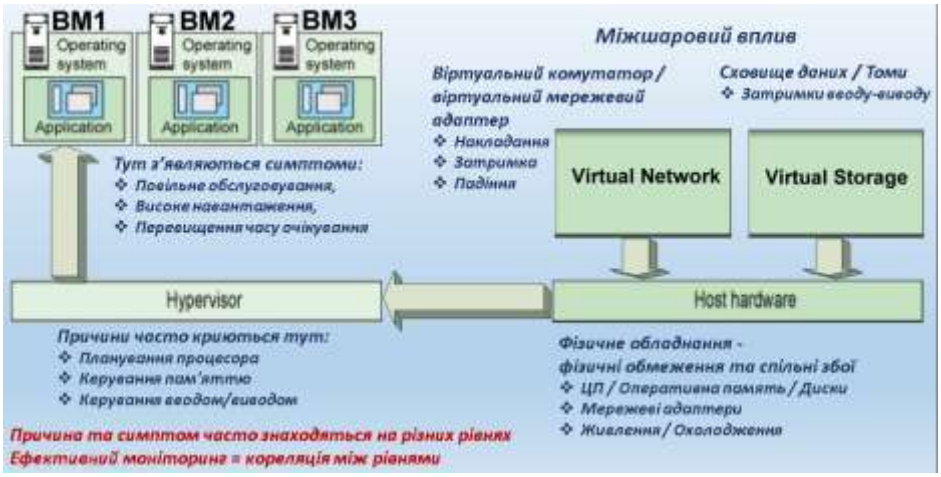


Рис.3.5. Віртуалізоване середовище як об'єкт моніторингу (єдина система, кілька рівнів).

Об'єкти моніторингу у віртуалізованих середовищах

Після того як ми розглянули архітектуру віртуалізованого середовища, логічно перейти від абстрактних рівнів до **конкретних об'єктів**, з якими працює система моніторингу. Саме на цьому етапі стає зрозуміло, що моніторинг у віртуалізації — це не один екран з графіками, а ціла мережа спостереження за різнорідними компонентами.

Кожен із цих об'єктів має власну роль у середовищі, власні типові метрики і власні сценарії відмов. І саме їх сукупність формує реальну картину стану інфраструктури.

➢ ***Фізичні хости***

Фізичні хости є базовими об'єктами моніторингу, навіть якщо у віртуалізованому середовищі вони не завжди перебувають у центрі уваги. Саме на них тримається вся логіка віртуалізації, і будь-яка проблема на цьому рівні має мультиплікативний ефект.

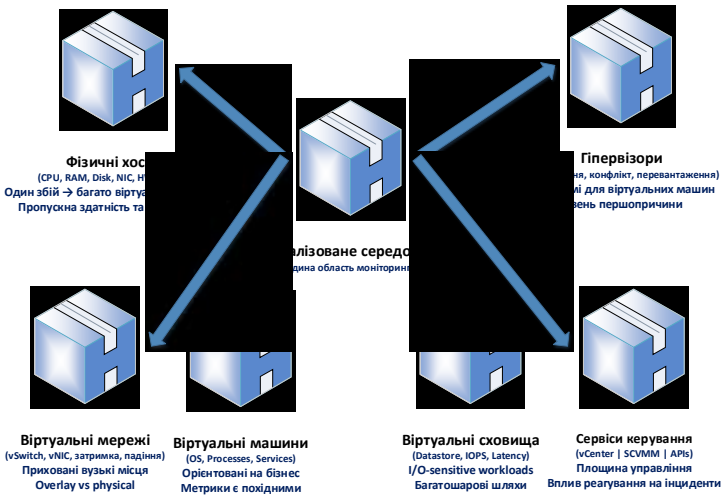


Рис.3.4. Об'єкти моніторингу у віртуалізованих середовищах.

Моніторинг фізичних хостів дозволяє оцінити, чи є у середовища достатній запас продуктивності і стабільності. Важливо не лише фіксувати аварійні стани, а й відстежувати поступову деградацію — зростання температур, збої дисків, нестабільність мережевих інтерфейсів. У віртуалізованому середовищі такі проблеми рідко проявляються миттєво, але з часом можуть призвести до серйозних наслідків.

➢ ***Гіпервізори***

Гіпервізор є центральним керуючим елементом віртуалізованого середовища і одним з найважливіших об'єктів моніторингу. Саме він визначає, як фізичні ресурси перетворюються на віртуальні, і як вони розподіляються між віртуальними машинами.

Моніторинг гіпервізора дозволяє побачити те, що залишається прихованим для операційних систем VM. Тут важливо відстежувати не лише загальну завантаженість, а й внутрішні процеси планування ресурсів. Саме на цьому рівні стає зрозуміло, чи справді проблема знаходиться у віртуальній машині, чи вона є наслідком загального перевантаження або конфлікту за ресурси.

➢ ***Віртуальні машини (VM)***

Віртуальні машини — це найбільш «видимий» об'єкт моніторингу, адже саме вони безпосередньо пов'язані з бізнес-сервісами. Для користувача або замовника сервісу не існує ні гіпервізора, ні фізичного хоста — існує лише VM, у якій працює потрібний застосунок.

Проте у віртуалізованому середовищі моніторинг VM має свої особливості. Важливо пам'ятати, що метрики віртуальної машини завжди є похідними. Вони відображають не лише внутрішній стан операційної системи, а й те, які ресурси фактично виділяє їй гіпервізор у конкретний момент часу. Тому моніторинг VM без урахування вищих і нижчих рівнів часто призводить до хибних висновків.

- ***Віртуальні мережі***

Віртуальні мережі — це той рівень, який часто залишається «за кадром», але водночас є одним з найкритичніших. Віртуальні комутатори, віртуальні мережеві інтерфейси та логічні сегменти мережі формують середовище взаємодії між віртуальними машинами.

Проблеми у віртуальній мережі можуть виглядати як збої застосунків або проблеми з продуктивністю, хоча фізична мережа при цьому працює стабільно. Саме тому моніторинг віртуальних мереж повинен бути тісно інтегрований з моніторингом гіпервізора і фізичних мережевих інтерфейсів. Без цього неможливо зрозуміти, де саме виникає вузьке місце.

- ***Віртуальні сховища***

Сховища у віртуалізованих середовищах є ще одним прикладом складної багаторівневої абстракції. Віртуальні машини працюють з логічними дисками, які фізично можуть бути розміщені на різних типах носіїв і систем зберігання.

Моніторинг віртуальних сховищ дозволяє оцінити реальну продуктивність введення-виведення і зрозуміти, як вона впливає на роботу VM. Часто саме тут ховаються причини повільної роботи баз даних або інших чутливих до I/O сервісів. Без спеціалізованого моніторингу цей рівень залишається «сірою зоною».

- ***Сервіси керування віртуалізованим середовищем***

Окрему категорію об'єктів моніторингу становлять сервіси керування — такі як VMware vCenter, Microsoft SCVMM або їхні аналоги. Хоча вони не беруть безпосередньої участі у виконанні обчислень, саме через них здійснюється управління, автоматизація і оркестрація середовища.

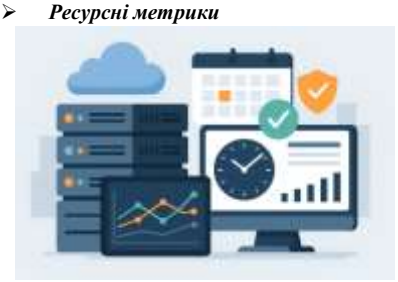
Втрата або нестабільна робота сервісів керування може не одразу зупинити віртуальні машини, але значно ускладнить реагування на інциденти, масштабування або відновлення після збоїв. Саме тому ці сервіси повинні розглядатися як критичні об'єкти моніторингу, навіть якщо вони не споживають значних ресурсів.

Ключові метрики та показники моніторингу у віртуалізованих середовищах

Коли ми говоримо про моніторинг віртуалізованих середовищ, дуже легко впасти в ілюзію повного контролю. Системи моніторингу здатні збирати тисячі метрик, будувати складні графіки й генерувати нескінченні звіти. Проте кількість даних ще не означає якість моніторингу. Ключовим є розуміння того, які саме показники дійсно відображають стан середовища, а які лише створюють інформаційний шум.

У віртуалізованих середовищах метрики набувають особливого змісту. Вони не просто показують поточне споживання ресурсів, а відображають складну взаємодію між фізичним обладнанням, гіпервізором і віртуальними машинами. Саме тому їх доцільно розглядати у кількох взаємопов'язаних групах.

- ***Ресурсні метрики***



Першою і найбільш очевидною групою є ресурсні метрики. Вони відповідають на базове питання: чи вистачає середовищу ресурсів для стабільної роботи. Проте у віртуалізації це питання значно складніше, ніж у фізичних системах.

Метрики процесора є показовим прикладом. Звичайне значення завантаженості CPU не дає повної картини. У віртуалізованих середовищах ключову роль відіграє overcommit — ситуація, коли сумарно виділені віртуальним машинам ресурси перевищують фізично доступні. Сам по собі overcommit не є проблемою, він є нормальним і навіть необхідним механізмом підвищення ефективності. Проблема починається тоді, коли з'являється CPU ready time або steal time — індикатори того, що віртуальні машини змушені чекати доступу до процесора. Саме ці показники часто пояснюють, чому система «гальмує», навіть якщо формально CPU не перевантажений.

Метрики оперативної пам'яті у віртуалізованих середовищах ще більш специфічні. Окрім звичних показників споживання пам'яті, тут з'являються такі механізми, як ballooning та swapping. Вони дозволяють гіпервізору перерозподіляти пам'ять між VM, але водночас є ознакою підвищеного тиску на ресурси. Memory pressure часто стає прихованою причиною деградації продуктивності, яку складно виявити без спеціалізованих метрик.

Дискові метрики у віртуалізованих середовищах також виходять за межі простого «швидко або повільно». Latency та IOPS є ключовими показниками, які дозволяють зрозуміти, чи справляється сховище з навантаженням. Особливо важливо те, що проблеми з дисковою підсистемою часто проявляються не на рівні сховища, а у вигляді затримок у роботі віртуальних машин і сервісів.

Мережеві метрики, такі як пропускна здатність і втрата пакетів, у віртуалізованих середовищах повинні аналізуватися з урахуванням логічної структури мережі. Проблема може виникати як на фізичному рівні, так і всередині віртуального комутатора, і без кореляції даних між цими рівнями її неможливо коректно інтерпретувати.

- ***Метрики стабільності та доступності***

Окрім ресурсів, важливим аспектом моніторингу є стабільність і доступність віртуалізованого середовища. Ці метрики відповідають на питання не «наскільки швидко», а наскільки надійно працює інфраструктура.

Статус віртуальних машин і хостів є базовим показником, але у віртуалізованих середовищах він має додаткові смисли. Наприклад, VM може бути запущена, але фактично не виконувати корисної роботи через проблеми з ресурсами або мережевими з'єднаннями. Тому простий статус «up/down» ніколи не повинен розглядатися ізольовано.

Особливу роль відіграють метрики міграції віртуальних машин — таких як vMotion або Live Migration. У нормальних умовах ці процеси є прозорими для користувачів і підвищують гнучкість середовища. Проте надмірна кількість міграцій або їхня тривалість може бути індикатором проблем із балансуванням навантаження або нестачею ресурсів.

Окремо варто виділити збої гіпервізора. Вони трапляються значно рідше, ніж проблеми всередині VM, але їхній вплив є значно серйознішим. Моніторинг повинен не лише фіксувати такі події, а й забезпечувати швидке виявлення їхніх причин і наслідків.

- ***Метрики ефективності використання ресурсів***

Третьою, але не менш важливою групою є метрики ефективності використання ресурсів. Вони дозволяють подивитися на віртуалізоване середовище не з точки зору інцидентів, а з точки зору раціонального управління.

Показники щільності VM допомагають оцінити, наскільки ефективно використовуються фізичні хости. Надто низька щільність означає перевитрати і неефективне використання обладнання, тоді як надто висока може призвести до нестабільності і зростання ризиків.

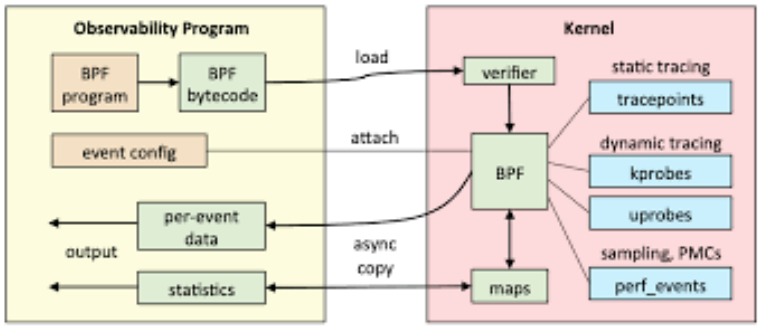
Метрики перевитрат і недовантаження дозволяють виявляти віртуальні машини, яким виділено значно більше ресурсів, ніж вони реально використовують, або навпаки — ті, що постійно працюють на межі можливостей. Саме ці дані часто стають основою для оптимізації середовища без додаткових інвестицій у «залізо».

Узагальненням цієї групи метрик є capacity planning — планування ємності. На основі історичних даних моніторинг дозволяє прогнозувати майбутні потреби у ресурсах і приймати зважені рішення щодо масштабування інфраструктури. Саме тут моніторинг перетворюється з реактивного інструменту на стратегічний.

Особливості та проблеми моніторингу віртуалізованих середовищ

Попри всі переваги віртуалізації, вона радикально ускладнює завдання моніторингу. Якщо у фізичних середовищах зв'язок між причиною і наслідком часто був прямим і очевидним, то у віртуалізованих інфраструктурах цей зв'язок стає опосередкованим і розмитим. Саме тут і виникають специфічні проблеми, які неможливо вирішити за допомогою традиційних підходів.

- ***Ефект "noisy neighbor"***



Однією з найхарактерніших проблем віртуалізованих середовищ є так званий ефект "noisy neighbor". Він виникає тоді, коли одна віртуальна машина починає активно споживати ресурси і непрямо впливає на роботу інших VM, що розміщені на тому ж фізичному хості.

З точки зору окремої віртуальної машини проблема може виглядати незрозумілою і навіть випадковою: продуктивність падає, затримки зростають, але внутрішні метрики не показують критичних перевантажень. Без моніторингу на рівні гіпервізора та хоста неможливо побачити справжню причину — конкуренцію за спільні ресурси. Саме тому ефект "noisy neighbor" є класичним прикладом проблеми, яку не видно зсередини VM.

- ***Overcommit ресурсів***

Overcommit ресурсів є фундаментальним механізмом віртуалізації і водночас джерелом багатьох проблем. Він дозволяє підвищити ефективність використання обладнання, але завжди працює на межі між оптимізацією і ризиком.

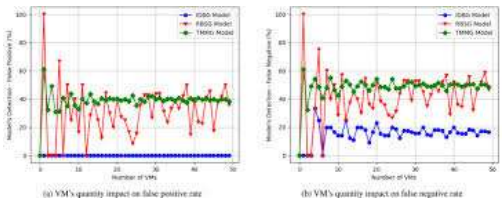
Поки реальне споживання ресурсів нижче фізичних можливостей, overcommit майже непомітний. Проте у моменти пікових навантажень він може призводити до затримок, деградації продуктивності або навіть відмов сервісів. Проблема ускладнюється тим, що класичні метрики завантаження не завжди сигналізують про небезпеку заздалегідь. Лише спеціалізовані показники, характерні для віртуалізації, дозволяють побачити, що середовище починає працювати у ризиковому режимі.

- ***Втрата прозорості між фізичним та логічним рівнем***

Однією з найсерйозніших концептуальних проблем віртуалізації є втрата прозорості. Логічні об'єкти — віртуальні машини, мережі, диски — відриваються від своєї фізичної реалізації. Для адміністратора або системи моніторингу стає складніше відповісти на просте питання: де саме фізично відбувається та чи інша операція.

У результаті одна й та сама проблема може виглядати по-різному залежно від точки спостереження. Те, що на рівні VM виглядає як нестача ресурсів, на фізичному рівні може бути наслідком перевантаження іншого компонента. Без чіткої кореляції між логічними і фізичними рівнями моніторинг перетворюється на набір розрізнених фрагментів, які складно зібрати в єдину картину.

- ***Динамічність середовища***



Віртуалізовані середовища за своєю природою є динамічними. Віртуальні машини можуть мігрувати між хостами, автоматично створюватися і видалятися, змінювати конфігурації у відповідь на навантаження. З точки зору експлуатації це величезна перевага, але для моніторингу — серйозний виклик.

Традиційні системи моніторингу часто будувалися на припущенні відносної статичності інфраструктури. У віртуалізації це припущення більше не працює. Моніторинг повинен бути здатним автоматично виявляти нові об'єкти, відслідковувати їх життєвий цикл і коректно інтерпретувати події, які раніше вважалися винятковими, наприклад міграції VM.

- ***Вплив моніторингу на продуктивність***

Ще однією, менш очевидною, але важливою проблемою є вплив самого моніторингу на продуктивність середовища. Збір великої кількості метрик, часті опитування та агентні підходи можуть створювати додаткове навантаження, особливо у середовищах з високою щільністю VM.

У віртуалізованих середовищах цей ефект посилюється, адже додаткове навантаження на одну VM або гіпервізор може впливати на сусідні системи. Тому під час побудови моніторингу важливо дотримуватися балансу між глибиною спостереження і його «вартістю» для продуктивності.

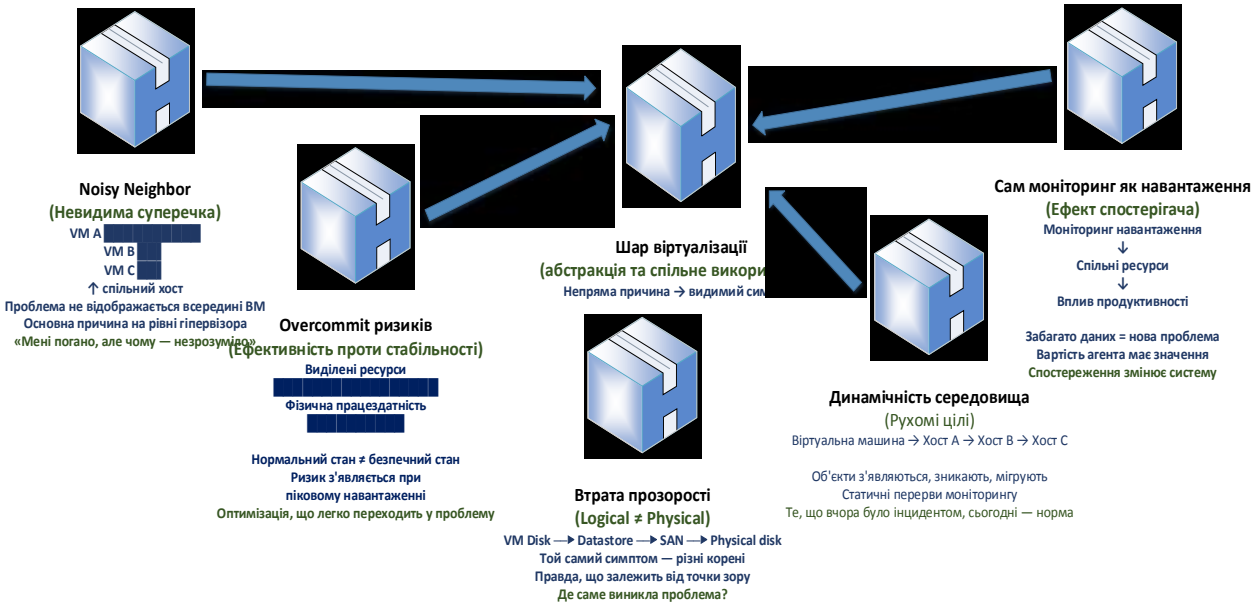


Рис.3.5. Віртуалізація розриває прямий зв'язок між причиною і симптомом.

Інструменти моніторингу віртуалізованих середовищ

Після розгляду метрик і проблем моніторингу виникає цілком природне запитання: якими саме засобами все це реалізується на практиці. У віртуалізованих середовищах вибір інструментів моніторингу ніколи не є нейтральним. Він визначає глибину видимості, якість кореляції даних і навіть стиль експлуатації інфраструктури.

Умовно всі інструменти моніторингу віртуалізації можна поділити на три великі групи: вбудовані засоби самих платформ, універсальні системи моніторингу та спеціалізовані комерційні рішення.

- ***Вбудовані засоби платформ віртуалізації***

Першим і найочевиднішим рівнем моніторингу є інструменти, які постачаються разом із самими платформами віртуалізації. Вони мають одну важливу перевагу — глибоку інтеграцію з внутрішньою архітектурою гіпервізора.

У середовищах VMware ключову роль відіграють засоби моніторингу vCenter та самих хостів ESXi. Вони дозволяють бачити специфічні для віртуалізації метрики, такі як CPU ready time, memory ballooning або детальну статистику міграцій VM. Саме ці показники часто недоступні або складно інтерпретуються у сторонніх системах.

У випадку Microsoft Hyper-V аналогічну роль виконують вбудовані інструменти Windows-екосистеми, які дозволяють спостерігати як за станом хостів, так і за віртуальними машинами. Тут важливо, що моніторинг тісно пов'язаний із загальною системою керування Windows-інфраструктурою.

Для середовищ на базі KVM ключовим компонентом є libvirt та пов'язані з ним інструменти. Вони надають доступ до низькорівневої інформації про стан віртуальних машин і гіпервізора, але часто вимагають більшої технічної підготовки для повноцінного використання.

Вбудовані засоби зазвичай добре підходять для первинного аналізу та оперативної діагностики, але рідко покривають всі потреби великої або гетерогенної інфраструктури.

- ***Універсальні системи моніторингу***

Наступний рівень — це універсальні системи моніторингу, які не прив'язані до конкретної платформи віртуалізації. Їх головною перевагою є єдиний підхід до спостереження за всією IT-інфраструктурою.

Такі системи, як Zabbix, Prometheus або Nagios, широко використовуються для моніторингу серверів, мереж, сервісів і застосунків. У контексті віртуалізації вони зазвичай працюють через плагіни, агентів або API гіпервізорів. Це дозволяє інтегрувати дані про віртуальні середовища у загальну картину стану IT-систем.

Однак універсальність має і зворотний бік. Такі системи часто не «розуміють» внутрішню логіку віртуалізації так глибоко, як нативні інструменти. Тому для повноцінного моніторингу вони потребують правильної конфігурації, додаткових модулів і чіткого розуміння того, які метрики дійсно є критичними.

Найкраще універсальні системи проявляють себе тоді, коли потрібно корелювати дані з різних середовищ: фізичних серверів, віртуалізації, контейнерів і прикладних сервісів.

- ***Комерційні спеціалізовані рішення***

Окрему категорію становлять комерційні рішення, спеціально орієнтовані на моніторинг і аналіз віртуалізованих середовищ. Їх ключова особливість — орієнтація не лише на спостереження, а й на аналітику та оптимізацію.

Такі платформи, як VMware Aria Operations, надають глибоке розуміння поведінки віртуалізованого середовища, включаючи автоматичний аналіз навантажень, виявлення аномалій і рекомендації щодо оптимізації ресурсів. Тут моніторинг фактично переходить у площину управління і планування.

Інші комерційні рішення, наприклад SolarWinds або PRTG, пропонують потужні засоби візуалізації, інтеграції та масштабування. Вони часто використовуються у великих організаціях, де важливими є централізований контроль, звітність і підтримка різнорідних платформ.

Основним недоліком таких рішень є їхня вартість і залежність від конкретного вендора. Проте у складних або критичних середовищах саме вони дозволяють зменшити операційні ризики і підвищити зрілість процесів експлуатації.

Підходи до побудови системи моніторингу віртуалізованих середовищ

Коли ми вже розуміємо, *що саме* потрібно моніторити і *які проблеми* характерні для віртуалізованих середовищ, виникає наступне, значно складніше питання: як правильно побудувати саму систему моніторингу. І тут немає універсального рішення, яке підійде всім. Підхід до моніторингу завжди є компромісом між глибиною спостереження, складністю впровадження та впливом на середовище.

- ***Agent-based та agentless підходи***

Одним з базових архітектурних рішень є вибір між агентним і безагентним моніторингом. Agent-based підхід передбачає встановлення спеціального програмного компонента безпосередньо у віртуальну машину або на хост. Такий агент має доступ до внутрішніх метрик операційної системи і може збирати дуже детальну інформацію про стан сервісів і застосунків.

Перевагою агентного підходу є глибина і точність даних. Проте у віртуалізованих середовищах він має і свої недоліки. Кожен агент споживає ресурси, ускладнює адміністрування і може стати додатковою точкою відмови. У середовищах з великою кількістю динамічних VM це створює значне операційне навантаження.

Agentless моніторинг, навпаки, намагається мінімізувати втручання у самі віртуальні машини. Дані збираються через мережеві протоколи, API або інтерфейси гіпервізора. Такий підхід краще масштабується і простіше підтримується, але часто обмежений з точки зору деталізації. Тому на практиці найчастіше використовується комбінований підхід, де агентний моніторинг застосовується лише для критичних VM або сервісів.

- ***Використання API гіпервізора***

Окрему і надзвичайно важливу роль у моніторингу віртуалізованих середовищ відіграють API гіпервізорів. Саме через них стає можливим отримання тієї інформації, яка принципово недоступна зсередини віртуальної машини.

API дозволяють бачити реальну картину розподілу ресурсів, стан міграцій, внутрішні черги планувальника і взаємодію між VM. Використання цих інтерфейсів є ключем до подолання втрати прозорості між фізичним і логічним рівнями, про яку ми говорили раніше.

Проте робота з API гіпервізора вимагає правильного проєктування. Некоректна частота опитувань або надмірна кількість запитів може створювати додаткове навантаження на керуючі сервіси. Тому моніторинг через API повинен бути не просто технічно можливим, а й грамотно спроєктованим.

- ***Централізований та розподілений моніторинг***

Ще одним ключовим архітектурним вибором є організація системи моніторингу — централізована чи розподілена. У централізованому підході всі дані збираються і обробляються в одному центрі. Це спрощує керування, візуалізацію і кореляцію подій, але створює потенційні точки перевантаження і відмови.

Розподілений моніторинг, навпаки, передбачає наявність кількох вузлів збору і обробки даних. У віртуалізованих і особливо гібридних середовищах такий підхід дозволяє зменшити затримки, підвищити відмовостійкість і краще масштабувати систему.

На практиці часто використовується гібридна модель: локальні компоненти збору метрик працюють ближче до середовища віртуалізації, а централізована система виконує агрегацію, аналіз і зберігання даних.

- ***Інтеграція метрик з логами та подіями***

Останнім, але надзвичайно важливим аспектом побудови системи моніторингу є інтеграція метрик з логами та подіями. У віртуалізованих середовищах багато проблем проявляються не поступово, а у вигляді окремих подій: збоїв, перезапусків, міграцій або помилок керування.

Метрики відповідають на питання «що відбувається», але часто не пояснюють «чому». Саме тут у гру вступає моніторинг подій і аналіз логів. Поєднання цих джерел інформації дозволяє будувати причинно-наслідкові зв'язки і значно скорочувати час реагування на інциденти.

У зрілих системах моніторингу метрики, події і логи розглядаються як єдине інформаційне поле. Це дозволяє перейти від реактивного реагування до проактивного управління станом віртуалізованого середовища.

Безпека та надійність моніторингу віртуалізованих середовищ

Коли система моніторингу починає глибоко інтегруватися у віртуалізоване середовище, вона неминуче перестає бути «стороннім інструментом». Вона отримує доступ до керуючих інтерфейсів, збирає чутливі дані і впливає на експлуатаційні процеси. Саме з цього моменту моніторинг стає не лише технічним засобом, а елементом безпеки та надійності всієї інфраструктури.

- ***Контроль доступу до API гіпервізора***

Одним із найбільш чутливих аспектів моніторингу у віртуалізованих середовищах є доступ до API гіпервізора. Через ці інтерфейси можна отримати детальну інформацію про стан хостів, віртуальних машин, мереж і сховищ. У багатьох випадках API також дозволяють виконувати керуючі дії — наприклад, запуск або зупинку VM.

Тому питання доступу до API не може розглядатися формально. Облікові записи, які використовуються системами моніторингу, повинні мати чітко обмежені права, відповідні принципу найменших привілеїв. Моніторинг повинен «бачити», але не «керувати», якщо у цьому немає прямої необхідності.

Окрему увагу слід приділяти зберіганню облікових даних і захисту каналів взаємодії з API. Компрометація таких облікових записів фактично відкриває шлях до всієї віртуалізованої інфраструктури, що робить їх критично важливими з точки зору безпеки.

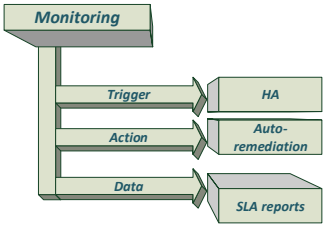
- ***Ризики компрометації через системи моніторингу***

Система моніторингу часто має доступ до великої кількості внутрішньої інформації: конфігурацій, топології середовища, імен хостів, мережевих адрес, а інколи й до логів або параметрів сервісів. Це робить її привабливою ціллю у разі атаки.

Якщо система моніторингу буде скомпрометована, зловмисник може отримати повну картину інфраструктури, що значно спрощує подальші дії. Саме тому моніторинг не можна розглядати як допоміжний сервіс з пониженими вимогами до захисту.

Практично це означає необхідність захисту самого моніторингового середовища: сегментації мережі, регулярного оновлення компонентів, контролю доступу до інтерфейсів керування і аудитів дій користувачів. У зрілих середовищах система моніторингу захищається за тими ж принципами, що й інші критичні сервіси.

- ***Вплив моніторингу на SLA та HA***



Окремого розгляду потребує вплив моніторингу на показники доступності та виконання Service Level Agreement (SLA). З одного боку, саме моніторинг дозволяє контролювати дотримання угод про рівень сервісу і швидко реагувати на інциденти. З іншого — некоректно побудований моніторинг може сам стати джерелом проблем. Нагадаю, що SLA відповідає на питання: «Наскільки добре сервіс повинен працювати?»

Надмірна кількість перевірок, агресивні опитування API або некоректно налаштовані агенти можуть створювати додаткове навантаження на гіпервізори і віртуальні машини. У високодоступних середовищах це може впливати на час відгуку, стабільність сервісів або навіть запускати небажані сценарії автоматичного перемикання.

Крім того, система моніторингу часто використовується як джерело сигналів для механізмів HA. Помилкові спрацювання або затримки у зборі даних можуть призводити до некоректних рішень — наприклад, до міграцій або перезапусків VM без реальної необхідності. Тому моніторинг повинен бути не лише функціональним, а й надійним та передбачуваним.

Практичні приклади та типові сценарії моніторингу

Після розгляду архітектури, метрик, проблем і інструментів моніторингу логічно перейти до практичних сценаріїв, з якими стикаються адміністратори віртуалізованих середовищ у щоденній роботі. Саме ці ситуації найкраще демонструють, чому моніторинг у віртуалізації не може бути фрагментарним і чому ізольований аналіз окремих метрик майже завжди призводить до помилкових висновків.

- ***Виявлення перевантаження фізичного хоста***

Один з найтиповіших сценаріїв — поступове або різке перевантаження фізичного хоста. Зовні проблема може проявлятися як загальна деградація продуктивності кількох віртуальних машин, які раніше працювали стабільно. Користувачі скаржаться на повільну роботу, але жодна окрема VM не виглядає критично перевантаженою.

У такій ситуації моніторинг на рівні гіпервізора дозволяє побачити реальну картину. Зростання CPU ready time, збільшення memory pressure або черг введення-виведення чітко вказують на те, що фізичний хост більше не справляється з поточним навантаженням. Саме тут стає очевидною роль кореляції метрик: лише поєднання даних з рівня VM і хоста дозволяє коректно ідентифікувати проблему.

- ***Аналіз деградації продуктивності віртуальної машини***

Ще один поширений сценарій — деградація продуктивності окремої віртуальної машини. На перший погляд здається, що проблема локальна: зростає час відгуку сервісу, збільшується навантаження на процесор або диск. Проте моніторинг внутрішніх метрик VM часто не дає однозначної відповіді.

У таких випадках ключову роль відіграє аналіз взаємодії VM з гіпервізором. Наприклад, низьке споживання CPU у поєднанні з високим CPU ready time свідчить про конкуренцію за ресурси, а не про проблему всередині операційної системи. Аналогічно, проблеми з дисковою продуктивністю можуть бути наслідком перевантаження спільного сховища, а не самої віртуальної машини. Саме цей сценарій добре ілюструє небезпеку ізольованого моніторингу.

- ***Моніторинг міграцій віртуальних машин***

Міграції віртуальних машин, такі як vMotion або Live Migration, зазвичай сприймаються як позитивна і навіть «непомітна» подія. Проте у практиці експлуатації вони можуть бути як симптомом проблем, так і їхнім джерелом.

Моніторинг дозволяє відстежувати частоту, тривалість і причини міграцій. Надмірна кількість таких подій часто свідчить про проблеми з балансуванням навантаження або нестачу ресурсів у кластері. Крім того, під час міграцій можуть тимчасово зростати затримки або навантаження на мережу і сховища. Без відповідного моніторингу ці ефекти залишаються непоміченими, але впливають на користувацький досвід.

- ***Реакція на збої гіпервізора***

Найбільш критичним сценарієм є збій гіпервізора або фізичного хоста. У такій ситуації відмова одного компонента може призвести до одночасної недоступності великої кількості віртуальних машин. Саме тут моніторинг переходить з режиму спостереження у режим підтримки відновлення.

Коректно налаштована система моніторингу дозволяє швидко зафіксувати сам факт збою, визначити його масштаб і відстежити процеси автоматичного відновлення, такі як перезапуск VM на інших хостах. Важливо, що моніторинг має не лише сигналізувати про проблему, а й підтверджувати успішність відновлення сервісів. Без цього неможливо оцінити реальний вплив інциденту на доступність і виконання SLA.

Рис.3.6. Практичні сценарії моніторингу віртуалізованих середовищ

Ці практичні сценарії добре демонструють головну ідею всієї лекції:
Моніторинг віртуалізованих середовищ — це не набір графіків, а інструмент розуміння поведінки складної, багаторівневої системи.