

ВИБІР ЦІЛЬОВОЇ ПЛАТФОРМИ ВИКОРИСТАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ АВТОМАТИЗОВАНОГО МАСШТАБУВАННЯ ХМАРНОГО ЗАСТОСУНКУ

Козачук Андрій

Вінницький національний технічний університет

Анотація

Здійснено вибір цільової платформи використання інформаційної системи автоматизованого масштабування хмарного застосунку. Було враховано такі фактори як простота розгортання та можливість програмного масштабування хмарних застосунків.

Визначено, що технологія Microsoft Azure App Service (Web apps) найкраще відповідає поставленим вимогам.

Abstract

Target platform for usage of informational system of automatic cloud application scaling has been chosen. Simplicity of deployment and abilities of scaling programming interface were used as main criteria

It was determined that Microsoft Azure App Service (Web apps) technology suites mentioned requirements the best.

Вступ

Сучасні платформи хмарних обчислень надають широкий спектр можливостей, що суттєво відрізняється в залежності від платформи. Політика ціноутворення також має свої особливості у кожного провайдера хмарних платформ. Тому актуальною є задача вибору оптимальної хмарної платформи для розміщення інформаційної системи масштабування хмарного застосунку.

Формат матеріалів доповіді

На сьогоднішній день поширення отримали три підходи до розміщення хмарних застосунків: інфраструктура як послуга (англ. infrastructure as a service, IaaS), що передбачає виділення певної кількості віртуальних машин у хмарі, що обслуговуються «повноцінною» операційною системою та взаємодіють між собою за допомогою засобів самого хмарного застосунку; платформа як послуга (англ. platform as a service, PaaS) передбачає наявність проміжного шару між віртуальними машинами та хмарним застосунком, що забезпечує відсутність прив'язаності до операційної системи та автоматичне балансування навантаження; програмне забезпечення як послуга (англ. software as a service SaaS) – при використанні цього підходу функціонування хмарного застосунку повністю забезпечується хмарною платформою. SaaS підхід не надає достатньої гнучкості налаштування, а PaaS не дозволяє використовувати всі переваги хмарної платформи без розробки складних механізмів балансування навантаження, використання PaaS підходу дозволяє використовувати всі переваги хмарних платформ без необхідності у розробці низькорівневих службових модулів, тому при аналізі хмарних платформ пріоритет надається саме PaaS платформам.

До сучасних хмарних платформ, що підтримують автоматичне масштабування хмарного застосунку можна віднести Amazon EC2 [1] та Elastic Beanstalk [2], Microsoft Azure [3], Google App Engine [4], GoGrid [5], Enstratus [6], також відомий як Dell cloud manager, та RightScale [7].

Розглянемо платформи розміщення хмарних застосунків, на базі яких може бути використана інформаційна система автоматизованого масштабування хмарного застосунку зі змінними піками навантаження. До найбільш популярних середовищ типу

“Платформа як послуга” (PaaS) належать Amazon AWS Elastic Beanstalk, Google App Engine, Microsoft Azure. Усі перераховані платформи можуть забезпечувати високу доступність при роботі хмарного застосунку за рахунок географічного розподілення серверів. При виборі конкретної платформи використовуються такі критерії як вартість хостингу та додаткові можливості платформи. З точки зору використання інформаційної технології масштабування хмарного застосунку необхідно розглянути можливості платформи по постачанню актуальної статистичної інформації про стан роботи хмарного застосунку та можливостей програмного інтерфейсу (API) масштабування.

Компанія Apprenda надає послуги хостингу PaaS-застосунків для корпоративних користувачів. Перевагою платформи Apprenda є велика кількість типів архітектур хмарних застосунків, що підтримуються цією платформою, що зумовлено її орієнтацією на корпоративний сегмент ринку хмарних обчислень. Платформа Apprenda надає додаткові механізми автентифікації та авторизації користувачів, що збільшує рівень захисту хмарного застосунку. Також наявний ряд .NET бібліотек, що спрощують розробку та розгортання хмарних застосунків. Недоліками платформи є обмеженість її використання індивідуальними користувачами та висока вартість абонплати.

Порівняння швидкодії та питомої вартості платформ [144] не виявило однозначного лідера – результати суттєво відрізнялись в залежності від конкретного тесту продуктивності, тому цей критерій не може бути використаний при виборі цільової платформи використання інформаційної системи автоматизованого масштабування хмарного застосунку зі змінними піками навантаження.

Google App Engine підтримує створення застосунків з використанням мов програмування Java, Python, PHP та Go. Розробка в межах цієї платформи ускладнюється необхідністю використання вузькоспеціалізованих рішень, які не працюють за межами платформи. Платформа підтримує три режими масштабування: ручне, при якому кількість віртуальних машин задається користувачем в файлі конфігурації; базове – при наявності запитів від користувачів активується віртуальна машина із застосунком, при відсутності запитів – хмарний застосунок переводиться в режим очікування; автоматичне – масштабування здійснюється за допомогою вбудованого модуля, що використовує час обробки мережевого запиту та частоту запитів для прийняття рішення щодо масштабування. У зв'язку з необхідністю використовувати жорстко прив'язані до платформи технології та незручністю автоматичного користувацького масштабування, використання Google App Engine в якості цільової платформи для інформаційної платформи масштабування є недоцільним.

Amazon AWS, зокрема служба Elastic Beanstalk дозволяє розміщувати застосунки розроблені за допомогою Java, Node.js, PHP, Python, Ruby, та ASP.NET. При цьому платформа забезпечує автоматичне оновлення програмного забезпечення та балансування навантаження між віртуальними машинами. Розгортання хмарного застосунку може бути здійснене з репозиторія GIT або з таких середовищ розробки як MS Visual Studio та Eclipse за допомогою спеціальних плагінів. Платформа Elastic Beanstalk підтримує автоматичне масштабування хмарних застосунків та масштабування за розкладом, проте не надає програмний інтерфейс для проведення масштабування, що робить її непридатною для використання з запропонованою інформаційною технологією масштабування хмарного застосунку.

Більш низькорівневе рішення від Amazon – Elastic Compute Cloud (EC2) має гнучку систему об'єднання віртуальних машин в групи масштабування, проте EC2 ближча до IaaS ніж до PaaS та вимагає виконання значних обсягів операцій для розгортання хмарного застосунку.

Платформа Microsoft Azure надає можливість розміщувати хмарні застосунки використовуючи три технології: Azure App Service (Web apps), Azure Cloud Services (Web roles) та віртуальні машини. Остання технологія відноситься до рівня IaaS та вимагає значних адміністраторських робіт по розгортанню та підтримці функціонування

хмарного застосунку. Розглянемо більш детально різницю між Azure Web apps та Azure Web roles, обидві з яких є PaaS технологіями.

Web apps надає можливість швидкого розгортання хмарного застосунку та зручного запуску планових задач за допомогою Azure WebJobs, в той же час дана технологія не підтримує деякі застарілі можливості ASP.NET та складні налаштування середовища виконання коду. З точки зору масштабування, Web apps надає зручний графічний інтерфейс та можливість автоматизованого масштабування за допомогою спеціальної утиліти. При масштабуванні не потрібно проводити повторну конфігурацію та розгортання хмарного застосунку, що важливо при наявності піків навантаження. Microsoft Azure Web pps дозволяє розмішувати ASP.Net та PHP застосунки, а також надає доступ до Java та Ruby SDK.

Web roles дозволяє проводити складні налаштування середовища, віддалено під'єднуватися до серверів, автоматично запускати задачі при старті сервера та використовувати необмежену кількість віртуальних машин. В той же час масштабування хмарного застосунку вгору вимагає його повторного розгортання, що значно ускладнює процедуру.

Таким чином, PaaS платформи можна розділити на низькорівневі (Microsoft Azure Web roles, Amazon EC2), що дозволяють проводити детальне налаштування середовища, в тому числі – проводити гнучке масштабування через програмні інтерфейси проте вимагають відносно великих затрат на розгортання та підтримку та високорівневі (Google App Engine, Amazon Elastic Beanstalk, Microsoft Azure Web apps), що дозволяють проводити швидке розгортання та мають вбудований механізм автоматичного масштабування.

Висновки

Таким чином, враховуючи такі особливості як простота розгортання та можливість програмного масштабування хмарних застосунків, що використовують технологію Microsoft Azure App Service (Web apps), інформаційну систему автоматизованого масштабування хмарного застосунку доцільно реалізовувати саме в контексті цієї технології.

Список використаних джерел:

1. Getting Started with Auto Scaling [Електронний ресурс] / Amazon – Режим доступа: <http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/GettingStartedTutorial.html>
2. Configuring Auto Scaling with Elastic Beanstalk. [Електронний ресурс] / Amazon. 2010 – Режим доступа: <http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/using-features.managing.as.html>
3. Портал Microsoft Azure [Електронний ресурс] / Microsoft – Режим доступа : <https://portal.azure.com/>
4. Scaling Based on CPU or Load Balancing Serving Capacity [Електронний ресурс] / Google Cloud Platform. – Режим доступа: <https://cloud.google.com/compute/docs/autoscaler/scaling-cpu-load-balancing>
5. GoGrid - A Datapipe Company. [Електронний ресурс] – Режим доступа: <https://www.datapipe.com/gogrid/>
6. Dell Cloud Manager: The Enterprise Cloud Management Solution [Електронний ресурс] – Режим доступа: <http://www.enstratus.com/home>
7. Ashalatha R. Evaluation of Auto Scaling and Load Balancing Features in Cloud / R. Ashalatha, J. Agarkhed// International Journal of Computer Applications. – 2015. – № 117(6). – Режим доступа: <http://research.ijcaonline.org/volume117/number6/pxc3902949.pdf>.