

Dapr i Tye – prostsze mikroserwisy .NET w chmurach i nie tylko

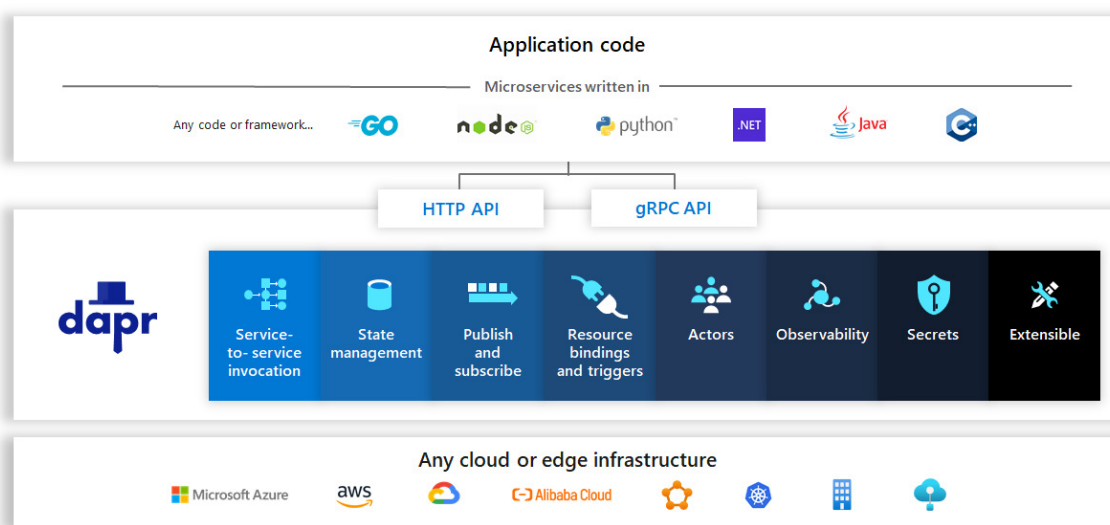
Czy nie chcielibyśmy łatwo przenosić naszych aplikacji mikroserwisowych między środowiskiem lokalnym a różnymi chmurami? Czy nie chcielibyśmy pisać mniej kodu, skupiając się jedynie na funkcjonalnościach biznesowych zamiast na różnych elementach infrastruktury, jak np. Kafka czy Redis? Czy nie życzylibyśmy sobie automatycznego tworzenia definicji obrazów Dockera i manifestów Kubernetes, ich wdrażania oraz prostego konfigurowania różnych narzędzi, jak np. Zipkin? Część z tych problemów rozwiązuje otwarta i niezależna od języka platforma Dapr. Tye to z kolei będące w fazie rozwoju narzędzie dla samego .NET. Zobaczmy, co może nam dać połączenie Tye z Dapr.

Zacznijmy od tego, że mimo nazwy .NET w tytule artykułu będzie on w sporej mierze przydatny dla każdego programisty nowoczesnych aplikacji rozproszonych zgodnych z postulatami Cloud Native. Do orkiestracji kontenerów wykorzystywać będziemy środowisko Kubernetes, które stało się wspólnym mianownikiem dla każdej chmury, rozwiązań *on-premise* czy też przetwarzania danych na krawędzi sieci w urządzeniach IoT. Spotykają się tu nie tylko światy chmur publicznych, tj. Microsoft Azure, Amazon Web Services (AWS) czy Google Cloud Platform (GCP), ale także Red Hat, VMWare czy OpenStack. W artykule poświęcimy całkiem sporo uwagi projektowi open source o nazwie Dapr, który jest otwarty na każdy język, natomiast dla .NET, JavaScript, C++, Python, Java i Go zawiera dodatkowe udogodnienia. .NET posłuży nam tutaj do napisania mikroserwisów w ramach przykładu. Programując już w .NET, przy okazji skorzystamy z narzędzia Tye, by nie tworzyć samemu plików dla Dockera i Kubernetes. I jeszcze uwaga dla osób śledzących

oficjalne wytyczne od Microsoft dla tworzenia aplikacji mikroserwisowych w .NET dostępne na stronie <https://docs.microsoft.com/en-us/dotnet/architecture/microservices/>, przybliżone także na łamach „Programisty” 8/2018 (75). W niniejszym artykule poruszona zostanie jedynie pewna część przedstawionej tam obszernej tematyki, skupiając się na zagadnieniach, które będzie można realizować prościej w nieco inny sposób.

I DAPR

Dapr (Distributed Application Runtime, patrz Rysunek 1) to przenośny, oparty na zdarzeniach runtime do szybkiego budowania odpornych, bezstanowych i stanowych aplikacji opartych na mikrousługach w chmurze i na krawędzi, pozwalający korzystać z dowolnych języków i frameworków. W swoim założeniu ma sprawić, byśmy mogli skupić się na biznesowej funkcjonalności, tworząc ją w ulubionym



Rysunek 1. Dapr (źródło: <https://github.com/dapr/docs/tree/master/overview>)