Microservices Simplified
Via Emerging App Platform
Frameworks

Author: C. Dunlap, Principal Analyst, Application Platforms, April 2017

Perspective

The next phase of software development goes beyond the cloud to address containerization, and microservices in particular. Microservices, especially as realized on cloud platforms, offer many advantages including polyglot programming, serverless architectures, as well as automated and uninterrupted patching and enhancement. DevOps and IT managers are demanding a more flexible application architecture to deliver robust, highly available applications. Over the coming year, we'll see new frameworks which let enterprises more easily build on the idea of containerization, taking what was previously a facet of service oriented architecture (SOA) and breaking that down into the smallest microservice components. With these components, IT and DevOps can independently develop and curate apps across the app lifecycle. More interestingly, they can also stitch those services into new apps where appropriate. The new tools and architectures built around microservices address not just the future but also the past. They address traditional Java EE multi-tier/n-tier architectures, which require more distributed service components and demand the use of automation to support continuous integration, continuous deployment (CICD).

Traditional application platform providers are well positioned to roll out such microservices strategies. They understand the importance of supporting mission-critical applications that scale and are easily maintained over time, leveraging portfolios (which include integration/connectivity), API management, and SOA solutions. Note, SOA supports the idea of reuse of broad application functions, while a microservices architecture allows the same at a much more granular level. Enterprises have been slow to move from cloud-based dev/test to workload production because the emerging architectures are extremely complex, and most organizations lack Netflix-scale programming expertise and resources in-house. Open source software technologies in particular will help drive microservices through container portability and adoption of various orchestration, monitoring, and push services.

Driving this evolution are increased demands being placed on IT operations and developers. In an API era where data is continually collected as a result of mobile and IoT endpoints, the cloud has become more important than ever for rapidly building apps and streaming services. Operations groups are tasked with determining where an app is run, whether it's a cloud-based service, in the data center as part of a virtual stack, private cloud or public cloud. Based on the efficiencies arising from server-level container infrastructures, such as virtual machines, these app development projects are increasingly being run within containers as microservices, which address operational issues around infrastructure portability and management. Operational issues are also prompting enterprises to rely on containers as a deployment option, helping ease management requirements associated with numerous microservices. PaaS’ evolution into a container-oriented set of microservices is in essence standardizing application platforms across disparate vendor technologies.
Beyond the rise of containers and microservices, there's a related developing trend around advanced automation, including Functions-as-a-Service (FaaS), also referred to as serverless architecture, which was popularized with AWS Lambda, with others following suit. The idea is simple but powerful. When an event is triggered a function is invoked automatically via a container to provide the context and execution framework for that event. This promises to automate and reduce IT operational overhead. A number of application platform vendors are working on projects within FaaS. These single automated provisioning/operations/-patching services include: AWS Lambda, IBM OpenWhisk, Microsoft Azure Functions, and Oracle Functions.

Again, some key open source projects currently underway such as IBM's OpenWhisk will help advance these market segments. Market leaders are investing in new initiatives for bringing popular development languages into modern development environments, namely the MicroProfile project, backed by IBM and Red Hat. This project creates a smaller subset JDK of Java EE focused on the cloud native space and ensures various deployment scenarios of microservices-based apps.

The year ahead is an important one for the industry as enterprises explore manageable containers to support microservices architectures, supported by innovative platforms evolving from middleware solutions and services. Go-to-market strategies among vendors have shifted as both IT operations and developers define how applications are built and deployed. These teams now face the daunting task of making sense of various technologies and strategies, all of which promise the best long-term approach for establishing hybrid and multi-cloud application development and deployment platforms.

See how IBM is approaching microservices and helping businesses build cloud native applications. Learn more: [http://ibm.co/buildmicroservices](http://ibm.co/buildmicroservices)