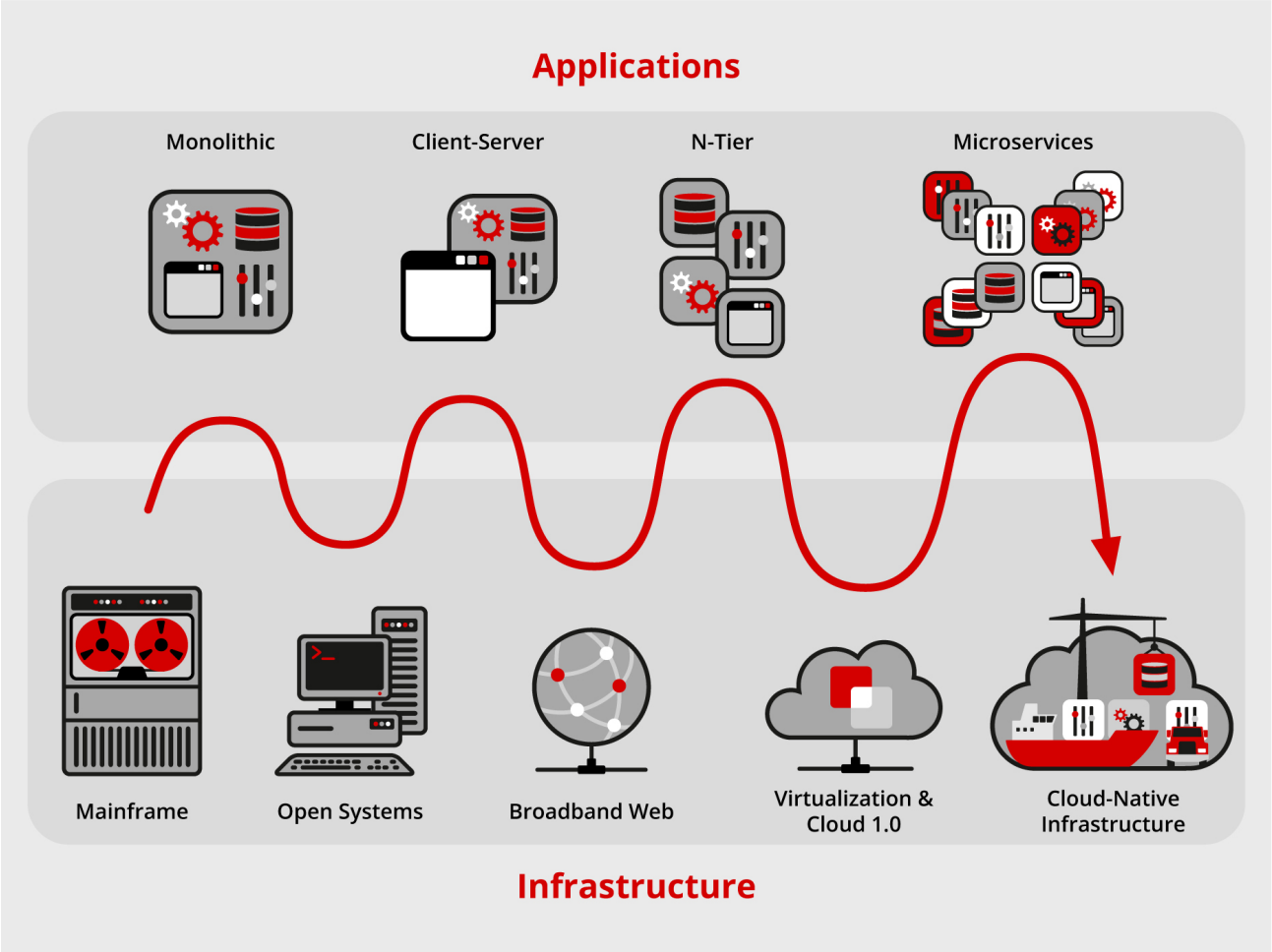
WING

Microservices, Containers And The Digital Dialectic

JANUARY 8, 2016 PETER WAGNER ESSAYS



In April 1956 the *SS Ideal X* set sail from Newark carrying 58 large metal boxes then known as “Trailer Vans”. The *Ideal X* was a modified WWII-era T2 tanker, and became the initial member of the world’s most successful container fleet. Soon the intermodal shipping container was standardized and purpose-built vessels—what we now call container ships—were created to carry this payload. Eventually loading cranes, trains,

trucks and even software and sensors were designed to optimize delivery of the standard container. The world's transportation infrastructure was revolutionized and global trade exploded.

This tale from the physical world is a vivid illustration of how workload innovations drive infrastructure innovations. The powerful cycle is even more dynamic in the digital realm, where changes in application architecture spawn transformations in technology infrastructure, and vice versa. At Wing, we call this cycle the "Digital Dialectic".

The latest evolution in this Dialectic revolves around containers composed of code rather than metal. These digital creations are part of a broader movement towards a radically different way of creating and deploying software, dubbed microservices, which involves deconstructing large applications into smaller, independent processes that interact via APIs. Containers provide a lightweight packaging mechanism in which microservices can run consistently without the need to deploy a hypervisor and incur the associated overhead.

Just as those big metal boxes revolutionized the world of trade, so the combination of containers and microservices will transform not only how enterprises develop applications, but also how they architect infrastructure and ultimately beat their competitors. Hundred-billion-dollar-plus markets encompassing the totality of data center infrastructure and management are up for grabs.

Origin story

Applications and infrastructure have been evolving in a kind of ping-pong relationship since the origins of IT. Changes in infrastructure driven by physics have made computing cheaper and faster. These shifts subsequently trigger changes in application architectures, which in turn motivate optimizations in infrastructure to support the new application paradigm.

The graphic above highlights how the Digital Dialectic evolved from mainframes and monolithic applications to open systems in the 1990s, which triggered an outpouring of

software products using the new client-server model of application design. The next phase saw the emergence of the web as a platform, which gave rise to N-tier application design with its web front-ends, business logic modules, app server middleware and powerful database back-ends. At the start of the new millennium the game changed as virtual machines burst onto the scene. Virtualization created a viable method to share hardware between different applications and laid the groundwork for public and private clouds.

The secret to virtualization's success was its transparency—a wholesale infrastructure transformation that explicitly required zero change or even awareness from application developers. But the victory was incomplete. Enterprise application development cycles still took months. It would fall to others outside the enterprise sphere to truly change the game. Web-scale giants such as Google and Amazon led the way, forced to meet their own needs for scale and speed via internal development and eventually serving the results up to the rest of the world in the form of the public cloud.

The Next Wave

What will the transition to a microservices-centric approach mean for the infrastructure landscape? The meteoric rise of Docker is a signal that an arms race is under way in the world of container technologies, where other contenders include CoreOS at the container engine level, and Mesos and Google's Kubernetes at the critical orchestration control point. This all sets the stage for a new wave of infrastructure innovations purpose-built for the emerging reality. At Wing, we use the term "cloud-native infrastructure" to describe these innovations. Here are some early examples from the primary infrastructure disciplines: compute, networking, storage and security.

Compute

Computation is the area most obviously affected by the new application paradigm. As applications are decomposed into services packaged in containers and scaled-out across many hosts, workload patterns change and bottlenecks shift. Google and Amazon have gone so far as to invest in custom silicon, seeking to offload some of their more intense,

recurring computations. Before long the microprocessor itself will include optimizations for microservices, just as previous generations carried virtualization-specific capabilities. APL Software, in which Wing is an investor, is one of a number of companies seeking to exploit the move to microservices, offering a technology that allows modern, single-process applications to scale seamlessly across multi-core processing infrastructure.

Networking

Networking across hosts, domains, geographies and organizational boundaries becomes increasingly important once distributed applications are pushed into production. Several companies have emerged to meet this challenge, and Docker itself has already acquired one (Socketplane). Other interesting work is being done in the Calico, Weave and Flocker projects.

Storage

Another set of missing capabilities has to do with state. Early users of containers were explicitly stateless, and some argue that this is where they are best suited. However, persistent storage of varying degrees of sophistication will be necessary to broaden the universe of applications that can be containerized. This will be particularly true for previously monolithic or N-tier applications being refactored, but new, greenfield environments will also have a similar need. Docker has already expanded its storage options in release 1.9 and startups such as Portworx and Datawise aim to offer even more.

Security

While containers offer many advantages, they also open up a new attack surface for hackers. Enterprises will need to ensure their containers are secure. They will also need to find the best ways to vet them with minimum interference to business and to ensure they and their contents adhere to security policies. Docker, Redhat, Blackduck and recent startups such as Twistlock and Banyan are focusing on the image; others are

focused on the runtime environment, aiming to ensure that distributed application components can be trusted, monitored and controlled—all according to policy and in a manner that meets the scale and latency requirements of a production workload.

Hyperconvergence

Once virtualization reached a certain level of maturity, the hyperconvergence movement gained steam, led by startups such as Nutanix and Simplivity. Incumbents got on board as well, with offerings like the VCE Vblock, NetApp Flexpod and VMware EVO:RAIL. It is much earlier for containerization, but several startups such as Datawise and Rancher are hoping the hyperconvergence wave will break faster this time around.

Eyes on the infrastructure prize

Prior phases of the Digital Dialectic have given rise to new sets of companies with large market capitalizations. This cycle will do so too. The opportunity is now for infrastructure innovators to step forward with cloud-native offerings that are purpose-built for microservices. It is in moments like this when the most value is created at the infrastructure level. We've seen it play it out numerous times before: the open systems movement; the buildout of the commercial Internet; the virtualization of enterprise IT…Now the fabric of the cloud itself is in play—and the stakes couldn't be larger.

---

http://wing.vc/perspectives/essays/microservices-containers-and-the-digital-dialectic
© 1990 — 2018 Wing, Menlo Park, CA