Cloud Native
Integrated Modern Engineering
As an early adopter, Deloitte invests significantly in Cloud Native Software Development because of the transformational impacts we’ve seen Cloud Native have on our client solutions and mission, our development teams and in the marketplace. As the number of organizations pursuing and adopting cloud has grown exponentially over the last decade, there is an increased focus on maximizing the scalability, resiliency, manageability and automation, observability, interoperability, and security that cloud computing offers whether it is within public, private, or hybrid cloud environments.

To help our teams and our customers achieve these objectives at scale, Deloitte established a dedicated Cloud Native practice under the Government and Public Services (GPS) Cloud Engineering Offering. Our operational definition of Cloud Native is fully aligned to that of the Cloud Native Computing Foundation (CNCF) (https://www.cncf.io/) and refers to solutions that are designed, developed, and operated in manners that take full advantage of the cloud computing model. Grounded in many of the same values as CNCF, our culture, team, products, and methods are guided from core principles that are focused on happier development teams, safer solutions and processes, better experiences, and faster delivery:

- **Happier:** Developers and teams are challenged and constantly learning, able to spend more time on core high value activities.
- **Safer:** Security is shifted left in the SDLC and is continuous throughout the lifecycle.
- **Better:** Changes users need don't take long (makes them happy and more mission productive).
- **Faster:** Automation and Independence allows applications to be developed, managed and deployed individually; The architecture also optimizes the benefits of cloud elasticity.
These principles guide how we approach and apply practices, methods, and capabilities and enable us to quickly deliver secure, quality software to our customers. We do this through a factory model that commoditizes common platform/components/tasks and provides reusable applications, components, and accelerators to jumpstart development. This model focuses on enabling mass production of cloud native solutions by reducing complexity and dependencies for software development teams. Here are several examples of key reusable accelerators we have incubated and matured to jumpstart internal development efforts as well as for our clients:

- **K8sUP** - Automated “push button” production grade Kubernetes container platform that is CNCF complaint and aligns with FISMA/FEDRAMP. AWS account cloud landing zone that builds a Kubernetes foundation (platform) for integrated Cloud Native CI/CD pipelines. These pipelines have standard software delivery practices built in for software quality, access/identify/secrets management, code and container repositories, security (to include VPN and MFA access), and operational monitoring. DevSecOps - integrated Cloud Native Customizable and reusable, Multi-cloud CI/CD Pipelines with software quality, chatOps and standard software delivery practices built in.

- **Microseeds** - Microservices Jumpstart providing pre-built microservices harvested after production authorization and refactored with additional documentation and various non-functional capabilities. Microseeds uses automation for faster authorization without trading quality or security.

- **Digital Armory** - Container hardening pipeline brings security-as-code from industry and government compliance; auto-remediation suggests and applies fixes when issues are found

- **DevEx Portal** - Provides developers with a Unified Developer Experience. Developers access defined templates, documentation, resources, and APIs to launch on factory.

Our approach is to focus on enabling at speed and scale, the building and shipping of secure products and providing the common platforms to “run” them. It is our goal to make these Cloud Native products and platforms repeatable, efficient, and accelerated through standards-based technology and patterns, architectural decisions integrated with plug and play capabilities. Ultimately reducing the noise, cognitive load, and complexities to enable teams to focus on delivering useful client solutions.
The challenges organizations face in adopting Cloud Native approaches and practices are not new to digital transformation. Like adopting any new software or technology, to overcome these challenges requires stakeholder buy-in achieved through education, engagement, outcomes, technical depth, and retooling:

- **Culture**
  - To many organizations, cloud is considered a destination where they would immediately realize benefits like cost savings, increased availability, etc. Organizations and teams need to understand that Cloud Native is a journey, not a destination. The journey is NOT about reaching a certain state but rather achieving desired outcomes and know where and how to improve through continuous visibility into key performance indicators and metrics.
  
  - Many organizations are still mentally (and emotionally) transitioning from data center IT Operations Management and getting familiar with operating/managing a cloud environment. Concepts like built-in security, continuous scanning, and automation are new and uncomfortable and without a tangible understanding of the practices and the value that they bring, the organization will be slow to adopt or even worse off forming bad habits and practices that will be costly and difficult to change downstream.
• **Platform**
  o Although there are managed services and software that helps with K8s orchestration and management, the reality is that designing, building, and operating production scale platform can be complex.
   - Platform engineering is crucial to building operations services that developers utilize for rapid app deployment. Organizations can benefit from independent development teams trained to use new platform services but need to invest in strong platform methodologies and collaboration between development and platform teams.
   - Moving to cloud enables opportunities for deep knowledge of the environment and deployments through CSP provided management plane. Effectively leveraging this management plane requires intentional platform configuration and training, but can lead to cost optimization, clarity in resource ownership, and full scope of the organization’s cloud landscape.

• **Skills Gaps**
  o When standing up a software product and/or platform team in a common software delivery model, many organizations will want to use their existing resources or apply junior resources to these teams without accounting for the skill (technical) roles needed to operate efficiently. If a gap analysis of skill roles is not done, these teams will struggle to deliver software effectively.
   - Examples of some key skill roles for a platform team include Infrastructure as Code, Kubernetes Containers, Platform architecture, security, and national standards.
   - To be successful, each persona (position) on a Software Delivery Team should have 2-3 skill roles and CLEAR responsibilities. Organizational leadership should qualify their teams to the furthest extent possible to ensure that team members are equipped with not only specialized skills to their primary duties, but secondary and tertiary skills to ensure coverage and the ability to operate on plan and on schedule when personnel are unavailable.
     - One person cannot or should not carry 8 skill roles or the team is imbalanced; 2-3 skill roles per persona is ideal.
The difference that Deloitte’s Cloud Native methodologies, assets and delivery models make to our teams and our clients is undeniable; Accelerating the development and delivery of secure software products improves mission outcomes.

Stay tuned for more information on our GPS Software Factory (SWF) assets and accelerators in the in future posts.