32 Principles and Practices to Successfully Transition to U.S. DoD Cloud Computing Data Centers

Tra·di·tion·al • Da·ta • Cen·ters (*trə-dĭsh'ə-nəl • dā'tə • sĕn'tərz'*). Centralized capital-intensive information technology (IT) assets, which include land, security fences, buildings, power-space-and-cooling, networks, computers, storage devices, operating systems, middleware, databases, development tools, and monitoring systems. Oftentimes, traditional IT applications were assembled one computer, server, rack elevation, or server farm at a time. That is, a computer system with a microprocessor, memory, motherboard, disk drives, network interfaces, operation systems, and input-output devices such as display screens, printers, and portable or removable tape, disk, or solid-state media. Traditional physical computer systems ranged in size from small to extremely large or monolithic. For instance, a computer system may be a laptop, desktop computer, Unix workstation, small rack elevation, small server farm with multiple racks, a building with many server farms, or even a monolithic collection of buildings with multiple data centers or high-performance computing equipment for massively parallel processing applications. About 80% to 90% of IT data centers are in the small to medium-sized range that sit underneath desks, or in conference room corners, hallway closets, or small conference rooms. Typically, small to medium-sized IT data centers were engineered to host a single information system or small ecosystem of highly interrelated applications. Rack elevations allowed engineers to assemble their computer systems one high-performance component at a time for high-performance computing needs, multitasking and multi-user systems, reliability and fault-tolerance, or fast network delivery. Building or campus-sized data centers often hosted larger server farms to simultaneously host many enterprise applications at once to achieve greater economies of scale.

Vir·tu·al·ized • Da·ta • Cen·ters (*vûr'choo̅-əl-īzd' • dā'tə • sĕn'tərz'*). A set of generalized, pooled, or shared information technology (IT) assets that are designed to transparently host many simultaneous enterprise applications. In a traditional IT data center, a large building may simply host many one-of-a-kind rack elevations or server farms each designed for the unique needs of individual information systems. Therefore, in its simplest case, the IT assets were not necessarily shared by many applications simultaneously. One server farm may be a storefront, another an enterprise storage repository or backup system, another, a high-performance computing system for running business analytics, and yet another for hosting generic application such as administrative, financial, or identity management systems. In virtualized data centers, all rack elevations or servers form one large farm, virtual hypervisors such as VMware are used to allocate some portion of compute power to individual information systems, and the applications are blind to the servers on which they are being hosted. Perhaps, 10% of compute capacity is allocated to administrative applications, 30% to compute-intensive business analytic systems, and 60% to storage repositories and backup systems. The pooled or shared microprocessors, memories, network devices, and storage systems may be bare-metal components or assets, and each completely self-contained virtual hypervisor contains its own version of an operating system, middleware, database, application suite, portal, or customer-facing storefront. Virtualized data centers are excellent for achieving fine-grained economies of scale, sharing IT assets between and among enterprise applications, and traditional single-threaded information systems with moderate compute and storage needs and extremely limited-scale performance capabilities.

Cloud • Com·put·ing • Da·ta • Cen·ters (*kloud • kəm-pyoo̅t'ĭng • dā'tə • sĕn'tərz'*). Cloud computing data centers are outsourced virtualized bare-metal data centers of which firms lease logical collections of IT assets running in hypervisor like containers to host their applications. For instance, a small enterprise may want a virtual IT data center to run an email, web, or accounting system server for its employees. It simply creates an account, allocates a small virtual server with microprocessors, a capable amount of memory, a network switch, gateway, and firewall, and some storage capabilities. The small enterprise then leases the virtualized IT data center on a per minute, hour, day, month, or annual basis. There are many advantages to leasing hypervisor-like IT data center assets to run enterprise applications. First of all, the small firm does not require its own building as the employees can simply work from home, the firm does not need an IT staff, the cloud service provider operates the underlying physical data center, etc. More importantly, risks of perimeter security, power-space-and-cooling, operations and maintenance, rapid technological obsolescence and entropy, and information security are completely transferred to the cloud service provider and bundled into the consumption charge. Furthermore, cloud service providers can pool their IT assets into massively parallel processing high-performance clusters, offer scalable databases and storage, and elastic on-demand or automatic scaling and fault-tolerance for serving millions of simultaneous customers, fluctuating demand and managing exabytes of data at a fraction of the cost of traditional IT data centers. They provide catalogues of assets, operating systems, middleware, databases, applications, and services to run in your environment. They can be configured in minutes from a laptop, tablet, or smartphone in one-click style.

Let's summarize some of these definitions and compare-and-contrast their attributes. Traditional data centers are designed to gain economies of scale by centralizing IT server farms, buildings, power-space-and-cooling, security, and IT maintenance. The downside is they are expensive to build, require years and billions of dollars, the servers are not shared amongst applications but are standalone, highly specialized server farms, and they are subject to rapid technological obsolescence and entropy. Virtualized data centers are also designed to gain economies of scale by centralizing IT servers farms, buildings, power-space-and-cooling, security, and maintenance. However, further economies of scale are gained by sharing generalized server farms, virtualizing individual environments for highly specialized information systems, and narrowing the number of IT personnel needed for operations and maintenance as their skills are simplified, generalized, and commonized. Cloud computing data centers are specially designed to gain monolithic economies of scale by centralizing IT servers farms, buildings, power-space-and-cooling, security, and maintenance for millions of simultaneous worldwide enterprises and reducing operations and maintenance costs to pennies on the dollar. They use sophisticated hypervisor like capabilities to allow enterprises to specially configure virtual IT data centers of all shapes and sizes for single applications, medium-sized ecosystems, large groups of enterprise systems, or extremely large massively parallel computing systems like social media applications with billions of simultaneous users storing and retrieving exabytes of data in fractions of a second. Cloud services are optimized for pooling microprocessors and memories into high performance massively parallel computers with exabyte scale databases and storage, and instant (automatic) on-demand elasticity for fluctuating demand, and clients are completely protected from technological obsolescence and entropy.

DoD Cloud Computing Principles & Practices

1. Lean-Agile • Think·ing (*lēn-ăj'əl • thĭng'kĭng*) Thin, slim, trim, light, fast; To apply small batches of WIP-limited business experiments to gradually tease out tacit, inexpressible requirements for innovatively new products and services

   ✓ Extremely small batches.
   ✓ Limited work in process (WIP).
   ✓ Frequent business experiments.

   The single most important principle of DoD cloud computing is lean-agile thinking. That is, teasing out tacit, hidden, and inexpressible customer, market, and end-user needs with small batches of WIP limited business experiments. Traditional thinking is based on predicting market, customer, and end-user needs for 5-10-or-15-years at time, and codifying these false assumptions in the form of integrated master schedules (IMSs), enterprise architectures, business requirements, etc. Furthermore, traditional thinking is based upon building traditional brick-n-mortar data centers one layer at a time (i.e., land, foundation, walls, roof, power, space, cooling, networks, racks, middleware, applications, security, etc.). Other than the market, customer, or end-user needs are incorrect, the technology is obsolete before you even order it from the catalogue, so your data center will be 20 to 25 years old before you complete it (if at all). Conversely, applying cloud computing principles, a few inexpensive business experiments, hypotheses, epics, and minimum viable products (MVPs) can be quickly formulated; an inexpensive on-demand virtual data center can be composed in minutes using a few keystrokes; a prototype can be placed in front of markets, customers, or end-users (i.e., warfighters); rapid feedback can be obtained to evaluate the hypotheses; and this cycle can be repeated many times in hours, days, and weeks without committing to an obsolete 5-10-or-15-year brick-n-mortar data center. Of course, it helps if you apply a lean intake system (i.e., Kanban), the batch sizes (i.e., scope) are kept very small, the work-in-process (WIP) or number of business experiments is also very small, and there is plenty of extra time to evaluate the results and make adjustments at a sustainable pace. The traditional IMS, enterprise architecture, and business requirements approach is built to cram as many unneeded requirements into an individual's daily schedule as possible, achieve full-utilization of all people on project, and build gold-plated, gold-fleeced, and over-scoped systems that are unnecessary, unneeded, and obsolete. It's a simple matter of optimizing large numbers of resources with full-utilization that actually increases vs. decreases risk (i.e., time, budgets, buildings, people, computers, networks, licenses, etc.). IT studies dating back to the 1980s show that over 95% of business requirements are not needed; chock full of unwarranted and defective market, customer, and end-user assumptions; and never used at all (if they are ever fielded). Traditional project plans represent enormous risk in terms of scope, budget, time, technology, performance, etc. With lean-agile thinking, most of these risks are mitigated, small numbers of tiny business experiments are rapidly implemented with virtual non-physical cloud computing assets; real, legitimate, and validated market, customer, and end-user needs are elicited; and the final product is rapidly composed using virtual non-physical cloud computing assets based on the bare minimum number of features.

2. Lean-Agile • Frame·work (*lēn-ăj'əl • frām'wûrk*) Support structure, skeletal enclosure, scaffolding platform, broad architecture; To apply a simple reference model to develop innovatively new products and services using lean-agile thinking principles

   ✓ Proven lean-agile framework or method.
   ✓ Embodies lean-agile values and principles.
   ✓ Cohesive system of ceremonies and practices.

   Another important principle of DoD cloud computing is to apply a lean-agile framework. Once again, when combining lean-agile thinking and cloud computing, a few inexpensive business experiments are quickly formulated; an inexpensive on-demand virtual data center is composed in minutes; MVPs are tested by market, customer, or end-user representatives (i.e., warfighters); and this cycle is repeated many times in hours, days, and weeks. But, like they say on television, "Don't try this at home, these stunts are performed by trained professionals!" Lean-agile frameworks are proven collections of highly cohesive principles, practices, roles, responsibilities, metrics, tools, and approaches to conduct successive business experiments in a highly structured manner. That is, they are lean, fast, efficient, and waste free, and most importantly, they have proven measurable properties. That is, if you're gonna compete in a Formula One race, you outta have a professionally designed sports car for this purpose. There's no sense in buying a jalopy and hopping it up or assembling a box car outta wood to run a Formula One race. That's simply nonsense. While small scale frameworks may help with lean workflow management like Kanban or implementation and execution like Scrum, few frameworks do much more than this. Complete enterprise frameworks like SAFe are replete with lean portfolio management (LPM), evolutionary architecture and design practices, business experimentation languages or notations, continuous delivery or DevOps practices, planning frameworks, and continuous improvement cycles at all levels. More importantly, lean-agile frameworks like SAFe are elastic and help you scale up and down as the complexity of the business experiment ecosystem grows, ranging from individual experiments to entire products, systems of systems, portfolios, and enterprises. The point is to do your homework, identify a small set of lean-agile frameworks, select one that is right for your context, and even consider one that can grow and scale with your business needs. Just some of the many lean-agile frameworks used for business experimentation include Scrum, Kanban, Design Sprints, 5x5 Business Experiments, Lean Startup, Startup Way, Scaled Agile Framework (SAFe), etc. When it comes to lean-agile frameworks, "Don't leave home without it," because they are essential to conducting business experiments with rapidly composable virtual non-physical cloud computing assets. For instance, examples of business experiments or hypotheses may include "Host and information repository in the cloud," "Evaluate a scalable database in the cloud," "Host an identity management system in the cloud," "Log, collect, and filter petabytes of Internet traffic," or "Host a geospatial intelligence system in the cloud." In all of these cases, these business experiments can be conducted in minutes, virtual non-physical data centers can be composed at fractions of a penny, and, most importantly, real mission requirements can be quickly identified.

3. Lean-Agile • Con·tract (*lēn-ăj'əl • kŏn'trăkt'*) Pact, deal, agreement, commitment, arrangement; To apply legally-binding terms and conditions for collaboratively developing innovatively new products and services using lean-agile thinking principles

   ✓ Legally-binding agreement.
   ✓ Based on lean-agile principles.
   ✓ Enforces collaborative behaviors.

   A critically important principle of DoD cloud computing is to form and apply lean-agile contracts. Basically, lean-agile contracts are legally-binding agreements between buyers and suppliers (i.e., government and a contractor or group of contractors) to apply lean-agile thinking principles AND a lean-agile framework. But, why is this so important, one may wonder? First of all, many government contracts involve the acquisition of IT-intensive systems hosted on traditional brick-n-mortar data centers or even virtual non-physical cloud computing platforms in the best case. Second of all, most government contracts are devoid of any lean-agile thinking nor lean-agile frameworks. Third of all, when a government contract does open the door for some limited lean-agile thinking or lean-agile frameworks and practices, it is also chock fully of hybridized traditional practices including 5-10-or-15-year IMSs, enterprise architectures, business requirements, and other outdated manufacturing practices from the last century. Of course, many, if not all government contracts cram as many unneeded requirements into an individual's daily schedule as possible, achieve full-utilization of all people on project, and build gold-plated, gold-fleeced, and over-scoped systems that are unnecessary, unneeded, and obsolete. In other words, most government contracts are set up to fail from the very beginning, usually by specifying traditional frameworks, hybrid traditional/lean-agile frameworks, or simply misapply lean-agile frameworks all together. In this latter case, a few leading government agencies have the vision to specify a lean-agile framework in the statement-of-work, but then use it to codify high-power distance buyer-supplier relationships, form large volumes of over-scoped requirements in a vacuum and then throw them over the wall into supplier or contractor lean-agile backlogs instantly freezing all queues, stopping all productivity, and bringing government acquisitions to a standstill. Conversely, lean-agile thinking, lean-agile contracts, and lean-agile frameworks are designed to "collaboratively" form small numbers of business experiments; rapidly implement them in virtual non-physical cloud computing servers; gather feedback from live market, customer, or end-user representatives (i.e., warfighters); and rinse-and-repeat many times at a sustainable pace until mission value is optimized. Lean-agile thinking is a mindset, culture, and belief system, it is not an IMS-driven contract to seek full-utilization of contractor personnel and acquisition failure. Therefore, lean-agile contracts are necessary for legally codifying, enforcing, and cultivating a culture of lean-agile thinking. When properly implemented, lean-agile contracts will be manageably sized, open, collaborative, team and consensus oriented, and based on shared responsibility and risk (vs. fear, blame, shame, and failure). Furthermore, lean-agile contracts are based on small batches of experiments, limited WIP, and a sustainable pace (vs. a false suicidal sense of full supplier or contractor utilization for efficiency's and insanity's sake).

4. Vir·tu·al • Ex·per·i·ments (*vûr'choo̅-əl • ĭk-spĕr'ə-məntz'*) Try, test, check, verify, investigate; To apply small batches of WIP limited hypothesis tests to gradually tease out tacit, inexpressible requirements for innovatively new products and services

   ✓ Small hypothesis tests.
   ✓ Minimum viable products (MVPs).
   ✓ Gathers fast, measurable market feedback.

   A key principle of DoD cloud computing is the use of virtual experiments. Let's start with some definitions of an experiment: 1) a systematic procedure carried out to support or refute a set of hypotheses or assumptions; 2) a scientific procedure undertaken to make a discovery, test a hypothesis, or demonstrate a known fact; or 3) a test, trial, procedure, or operation to discover something unknown or evaluate a hypothesis. From these definitions, we gather that there are some unknowns, uncertainties, risks, assumptions, or disputed facts. A traditional business analyst or systems engineer may compose a document with hundreds or thousands of unknowns, uncertainties, risks, assumptions, or disputed facts and falsely call these "testable business or systems requirements." Well, if you build a house on sand, it will most certainly collapse, and upwards of 90% of traditional programs and projects fail each year worldwide for precisely these reasons (i.e., false assertions, business requirements, systems requirements, etc.). Failed traditional projects cost worldwide enterprises, organizations, and businesses $4 to $5 trillion in lost investments each year. Conversely, business experiments are based on a few simple hypotheses to evaluate assumptions, they are narrowly scoped (i.e., can be evaluated in hours, days, and weeks), they are inexpensive (i.e., because they are small and narrowly scoped, or because they are composed in quick and inexpensive virtual non-physical cloud computing environments), and they are designed to quickly gather measurable feedback, facts, and data from market, customer, and end-user representatives (i.e., warfighters). Let's try to put this into context with a quick and easy example. Let's say a military agency (i.e., warfighters) are deployed to a mission theater (i.e., battlefield) and have a need to quickly gather some form of military intelligence (i.e., enemy numbers, strength, assets, positions, movements, radio signals, etc.). Perhaps, through the chain of command, a general asks for a new information system to gather such data. In this scenario, a government agency may ask for a few hundred million dollars to compose a new military information system in 5-10-or-15-years using IMSs, enterprise architectures, and business requirements. Instead, a lean-agile innovator, may suggest rapidly fielding an existing information system for this purpose or quickly composing a simple new one using virtual non-physical cloud computing assets in hours, days, and weeks. Furthermore, warfighters can immediately deploy the new information system while the battle is still going on (instead of waiting 5-10-or-15-years for an IMS to complete). Maybe the suggested legacy system is ineffective, or the prototype needs some more bells and whistles. In this latter scenario, lean-agile innovators rapidly field a series of small, inexpensive, virtual experiments in a matter of hours and days until warfighter needs are satisfied, an upper hand is gained in the battlefield, a better defensive posture is secured, and losses are minimized.

5. One-Team • Cul·ture (*wŭn'tēm • kŭl'chər*) Organizational ideals, values, norms, laws, mores; To create an open, collaborative, cooperative, communicative, and teamwork-oriented environment of shared responsibility with little to no power-distance

- ✓ Teamwork oriented principles.
- ✓ Extremely low power-distance.
- ✓ Shared collaborative responsibility.

An essential principle of DoD cloud computing is a one-team culture. But what exactly does this mean? Well, it's pretty obvious (i.e., the entire group of buyers and suppliers behaves and performs as a single team or team of teams)! In other words, buyers and suppliers behave as though they had equal authority, decision making rights, and shared responsibility for successes and failures (i.e., outcomes). Deep divisions between buyers and suppliers have existed for millennia. Typically, the emperor, king, or state had all of the resources, so they represented the buyer. The subjects of the emperor, king, or state represented the craftsman, skilled workforce, or labor to convert the resources into finished goods and services, so they represented they supplier. Think of King Nebuchadnezzar demanding that his satraps read his mind and interpret his dream under the threat of death. In this case, clearly the buyer has all of the power. However, this isn't always the case. Sometimes the supplier has all of the power. Think of an energy firm that has a monopoly on the market. In this case, the energy company has all of the power and can sell their energy at any price or (inferior) level of service. When the power of the buyers and suppliers is unbalanced or unequal, then someone will eventually come out on the short end of the deal (i.e., without a head or electricity during the coldest or hottest time of the year). As the old saying goes, "Power tends to corrupt, but absolute power corrupts absolutely!" In most government acquisitions, the buyer or government agency has ALL of the power, so they are entitled to form ill-constructed contracts using manufacturing principles from the last century or hybridized contracts with a pantheon of ineffective practices, metrics, tools, and technologies. Furthermore, since the government has all of the power, the supplier is at the mercy of the ineffective acquisition contract to satisfy the acquisition outcomes. It is simply a master-slave relationship! However, the imbalance doesn't stop with the relationship between the buyers (government) and suppliers (contractors). Oftentimes, the supplier (contractor) is in its own buyer-supplier arrangement (i.e., there is a prime contractor or integrator with a pantheon of small, medium, and large subcontractors). This ecosystem of buyer-supplier relationships between the government and contractors, and between the contractors themselves, leads to a deep set of divisiveness, lack of collaboration and cooperation, and, most certainly, little to no teamwork at all (not to mention unmanageable scale, scope, size, and complexity). At the heart of lean-agile thinking is the notion of highly collaborative cross-functional teams. That is, no one constituency, be it government or contractor, or prime contractor or subcontractor, as all of the answers, so they must collaborate, cooperate, and communicate to formulate a solution together. It's not a divide and conquer approach, where its components have been functionally decomposed into its constituent parts that can be built in a vacuum and assembled later.

6. Se·cu·ri·ty • First • Mind·set (*sĭ-kyo͝or'ĭ-tē • fûrst • mīnd'sĕt'*) Guards, defenses, assurance, protections, precautions; To plan for and build in information security principles early and often when creating innovatively new products and services

- ✓ Application security lifecycles.
- ✓ Application security practices and tools.
- ✓ Application security technology, evaluation, and testing.

An immutable principle of DoD cloud computing is a security first mindset. That is, the application of security values, principles, frameworks, practices, metrics, tools, and technologies must take precedence from day one instead of as an afterthought (i.e., bake-it-in vs. ice-it-on)! Each cloud computing initiative, program, or project should have a designated team of trained, certified, and experienced security engineers to oversee its implementation. The security team will establish a security framework replete with acceptable practices, metrics, tools, and technologies to ensure the cloud computing implementation has adequate guards, defenses, protections, and mitigation strategies. The security team is responsible for training and certifying the engineers in the security framework, overseeing and evaluating its security posture throughout its lifecycle, and evaluating and certifying the final implementation. Whether or not the cloud computing implementation is a small business experiment, series of small business experiments, or a final production system, the security team will help establish security requirements, architectures, designs, tests, certification procedures, monitoring guidelines, and mitigation plans. Sometimes production data and assets are used in business experiments or business experiments are used as production systems. Therefore, business experiments should be subject to some level of security principles, practices, tools, metrics, technologies, tests, certifications, and other safeguards. Apart from applying traditional, lean, and agile security lifecycles, principles, and practices, the use of virtual non-physical cloud computing assets provides many options for controlling, managing, or even mishandling data center, application, and information security. Typical assets may include virtual storage, compute or processing servers, load balancers, gateways, identity management, switches, routers, gateways, firewalls, etc. Each of these assets come with dozens or hundreds of security switches and access control privileges. The biggest mistake a cloud computing developer can make is to open all of the privileges for business experimentation, exploratory, and prototyping purposes, and then leave these switches open if the MVP is migrated or elevated to a production system (even for canary testing purposes). The last thing you wanna do is expose an unsecure multi-petabyte information repository of sensitive market, customer, or end-user data into the wild. There is an army of international hackers eagerly awaiting an opportunity to grab your data if even one of these security switches is found in the wrong (open) position. Sometimes, cloud computing identity management controls or credentials are built into the cloud computing assets themselves (i.e., access to a front-end website, grants access to a gateway, router, processor, or storage system implicitly, so only one of the assets in the cloud computing supply chain has to be compromised in order to access the entire cloud computing implementation smorgasbord.

7. Com·mer·cial • Clouds (*kə-mûr'shəl • kloudz'*) Open, public, market, industry, commodity; To acquire the use of open, publicly available cloud computing resources, applications, and technologies for creating innovatively new products and services

- ✓ Use of public cloud computing.
- ✓ Use of public platforms and services.

✓ Use of public platforms to host applications.

An important principle of DoD cloud computing is the use of commercial clouds. That is, DoD projects should utilize commercial cloud service providers, i.e., Amazon Web Services (AWS), Microsoft Azure, Google Cloud, IBM Cloud, Oracle, Rackspace, VMWare, etc. Commercial cloud service providers are just as reliable and secure for Fortune 500 firms as they are for most U.S. DoD systems and services. Use of commercial cloud service providers offer numerous benefits such as economies of scale, widest global availability zones, reliability, maintainability, scalability, elasticity, fault tolerance, and security. Most importantly, commercial cloud service providers offer a vast number of application technologies such as commercial application licenses, scalable databases, DevOps ecosystems, and provide instant protection from the inexorable march of technological obsolescence. All a DoD systems designer has to do is rapidly compose a virtual non-physical data center consisting of a highly secure network fabric, processing and storage capability, and cooperative ecosystem of commercial applications, business experiments, and legacy systems. Better yet, cloud computing systems can be rapidly composed, configured, operated, maintained, and even disposed of in minutes and seconds from mobile platforms such as laptops, tablets, smartphones, and other personal devices. Gone are the days when it requires a building, group of buildings, or geographically distributed collection of buildings to operate a DoD IT data center fabric to enforce and ensure continuity of operations (COOP) policy, mitigation strategy, or plan. COOP is "built-in" to commercial cloud service providers from the get-go and no capital investment in brick-n-mortar facilities is necessary. DoD data centers are plagued by expensive real estate, security perimeters (gates) and security guards, proximity to commercial and DoD communication links (high-bandwidth fiber optic cables), scalability of the physical footprint to support future growth, and, of course, rapid technology obsolescence. With commercial cloud service providers, all of these risks, costs, and parameters are absorbed in a per minute or per hour usage rate. DoD cloud projects can spin up a virtual cloud in minutes, conduct a mission experiment, and take it down for a few dollars (i.e., its completely unnecessary to beg agency decision makers for billions of dollars to build a one-time failed mission experiment—Traditional capital-intensive IT brick-n-mortar data center). In the worst case, DoD agencies can simply ask cloud service providers to apportion a block of data centers exclusively for DoD use in order to create a physical perimeter between DoD and non-DoD data center assets, clients, systems, and networks. In this case, the cloud service provider still apportions, builds, operates, and maintains the DoD cloud as though it was a commercial cloud, replete with the virtual, non-physical cloud computing products and services. This sacrifices economies of scale a bit but has many upsides too.

8. Com·mu·ni·ty • Clouds (*kə-myo͞o′nĭ-tē • kloudz′*) Group, league, association, organization, confederation; To form, apply, and utilize shared cloud computing resources, applications, and technologies for creating innovatively new products and services

✓ Use of shared cloud computing.
✓ Use of shared platforms and services.
✓ Use of shared platforms to host applications.

An obvious principle of DoD cloud computing is the formation of community clouds. That is, leveraging economies of scale by pooling DoD resources for common community cloud data computing data centers. There are dozens of major DoD agencies and hundreds if not thousands of individual departments. Furthermore, taxpayer resources are not equally apportioned to these agencies and departments, and only a small handful of them have the lion's share of the annual DoD budget. When it comes to IT budgets to fund brick-n-mortar IT data centers, custom clouds, on-premises clouds, or acquisition of commercial cloud services, only a small handful have the IT budgets to do so. Over 90% of DoD agencies and departments have little to no IT budget at all, therefore, it makes sense for the DoD to pool its IT resources and acquire, build, or host community or shared cloud services. For the most part, the use of commercial service providers is more than adequate for 80% to 90% of DoD information processing needs (i.e., routine administrative services). Perhaps, another 5% to 9% have some level of information security requiring the formation of commercial cloud services in on-premises or specially apportioned data centers. In very unique cases, there are specialized communities within the DoD with higher levels of information security requirements than normal, which require the placement of commercial cloud services in highly specialized on-premises data centers for highly sensitive information processing needs. The point is, rather than ask hundreds of DoD agencies and departments to standup individual brick-n-mortar IT data centers, host on-premises commercial cloud services, or simply use commercial cloud services outright, it may be advantageous to pool resources for similar communities, business or mission functions, or even highly sensitive information processing needs. It may be possible to build an ecosystem of common IT data centers to host commercial cloud service technologies and simply allocate firewalled apportionments of these assets to administrative, business, common mission, or specialized information processing needs (i.e., build a large common DoD cloud). However, the DoD should reduce the dependence upon on premises IT data centers to those which absolutely need physical separation of IT assets and information from public cloud service providers. That is, as has already been suggested, 80% to 90% of DoD information processing needs can be handled using cloud services providers without on-premises hosting of IT assets, applications, and information processing systems, repositories, and databases. For the most part, the DoD has gotten out of the business of specifying IT delivery frameworks, methods, and tools, and should likewise get out of the business of building IT brick-n-mortar data centers, custom clouds, and on-premises clouds, because it is simply a waste of taxpayer resources due to rapid technological obsolescence and exponentially increasing security vulnerabilities.

9. Re·gion·al • Clouds (*rē′jə-nəl • kloudz′*) Area, place, country, continent, hemisphere; To form, apply, and utilize geographically distributed cloud computing resources, applications, and technologies for creating innovatively new products and services

✓ Use of geographically distributed clouds.
✓ Use of geographically distributed platforms and services.
✓ Use of geographically distributed platforms to host applications.

Another major principle of DoD cloud computing is the formation of regional clouds. For instance, DoD agencies and

departments are domestically and geographically distributed for a variety of reasons. Some may be near large military air bases, test ranges, storage depots, remote defensive positions, hardened underground facilities, or forward deployed to remote parts of the world for strategic and tactical reasons. Perhaps, the purpose of an international location is to protect an ally, have access to strategic and tactical touch points, protect a volatile region or valuable market, or simply curtail illegal activity that compromises global markets and national security. In these cases, it may be necessary and even prudent to build globally distributed regional clouds for use by these distributed DoD forward deployed sites. Even commercial providers forward deploy regional clouds for a variety of information processing purposes. These are often referred to as "availability zones," that offer localized storage; specialized IT assets, applications, and internationalization services; access to high-traffic markets, metropolitan areas, and consumer populations; global fault tolerance and redundancy (i.e., continuity of operations); and, more importantly, "low latency!" That is, it makes no sense if your IT data center is in the middle of one country while the markets, customers, and end-users (i.e., warfighters) are in another region, country, geographical location, forward deployed base, or active military operations theater (i.e., battlefield). Without regional clouds, system performance is slow, unreliable, and subject to network outages; reliability, availability, and maintainability is sacrificed if a centralized cloud suffers a system outage; and, of course, the entire worldwide DoD posture is compromised if the centralized cloud is hacked, damaged, or otherwise physically disabled. So, there are many upsides to building "availability zones," geographically distributed IT data centers, or regional clouds. AWS has about 50+ "availability zones" or regional clouds and Microsoft boasts of having 100+ "availability zones" or regional clouds. Again, it makes sense for geographically distributed DoD agencies and departments to take advantage of the "availability zones" and regional clouds of commercial cloud service providers, especially for routine administrative processing. Doing so provides DoD agencies and departments instant access to a global fabric of state-of-the-art cloud computing services without large-scaled capital-intensive systems taking decades and billions of dollars to complete. Furthermore, by leveraging the "availability zones" and regional clouds of commercial cloud service providers, DoD agencies and departments instantly thwart the inexorable and exponential onset of technology obsolescence and uncontrollable entropy. That is, commercial cloud service providers constantly provision their "availability zones" and regional clouds with state-of-the-art IT processing, storage, networking, application, and security capabilities. DoD agencies simply can't compete.

10. Mis·sion • Clouds (*mĭsh'ən • kloudz'*) Goal, charge, purpose, operation, assignment; To form, apply, and utilize specialized cloud computing resources, applications, and technologies for creating innovatively new products and services

- ✓ Use of specialized cloud computing.
- ✓ Use of specialized platforms and services.
- ✓ Use of specialized platforms to host applications.

A critically important DoD cloud computing principle is the ability to rapidly compose and deploy tactical mission clouds. That is, a war, battle, or military operation ensues, and the battle units need IT data assets in the field quickly to host and utilize mission systems to prosecute the battle. In traditional scenarios, portable IT data centers may be rapidly deployed, configured, stood up, and operated to host IT mission information systems. However, there are numerous limitations to these solutions. For one, they have limited processing, storage, scalability, and reliability capabilities. Many DoD mission applications are processing and storage hogs, and portable IT data center assets offer limited warfighting capabilities at best. They require immense power, space, and cooling; they emit large radio noise signatures that compromise battle unit positions; and, if they fail or are attacked, then the field units are left with little to no information processing capabilities to support battlefield operations. With the use of commercial or on-premises commercial cloud service providers, this obviates the need for heavy-duty physical IT data center assets in the field. DoD information systems can be hosted in geographically distributed "availability zones" and regional clouds where state-of-the-art, elastic, fault-tolerant, and high-performance processors, storage, and high-bandwidth communication systems exist, which can be simply accessed through local smartphones, tablets, laptops, and other lightweight thin clients and IT devices. Think of the National Football League (NFL)—A U.S. football team is essentially a battlefield, tactical plans emerge from play-to-play, and teams of coaches use portable mobile devices like Microsoft Surface tablets to access high-performance information processing systems hosted on commercial cloud service providers in real-time. That is, high-capacity video is captured in state-of-the-art cameras, transmitted and stored in commercial clouds, and then accessed and replayed over and over again on portable tablets on the field during the game to determine what went well, what is going bad, and what to do better on the next play. Access to these "availability zones" or regional clouds from the battlefield allows the coaching teams to make real-time battlefield adjustments from portable tablets and increase the probability of winning the game, while opponents are doing the same. The worst-case scenario would be for an NFL team to have a traditional brick-n-mortar IT data center subject to constant outages and security vulnerabilities, low bandwidth communications, and, of course, limited processing and storage capacity, scalability, elasticity, and fault tolerance. Billions of dollars are at stake in an NFL game, battlefield injuries change the course of the ebb and flow, and, of course, standings, playoffs, and the grand prize of a Super Bowl is on-the-line. There's no time for mistakes, there's no place for a traditional brick-n-mortar IT data center, and there's no time for a system outage or security compromise.

11. Vir·tu·al • Da·ta • Cen·ters (*vûr'choo̅-əl • dā'tə • sĕn'tərz'*) Servers, storage, networks, processors, applications; To apply non-physical cloud computing resources, applications, and technologies for creating innovatively new products and services

- ✓ Use of non-physical IT data centers.
- ✓ Use of non-physical IT platforms and services.
- ✓ Use of non-physical IT platforms to host applications.

The essence of DoD cloud computing is the principle of automatically composing virtual data centers. This means exactly what it says, composing one-touch virtual data centers, perhaps using a few keystrokes, or even composing them with a scripting language like Javascript. Basically, commercial cloud service providers have assembled large collections of physical

data centers throughout the world with almost every conceivable technology asset, option, device, or application. This includes processors (compute power), microprocessors, memory, networks, switches, routers, load balancers, storage devices, operating systems, middleware components, databases, applications, etc. Using a remote desktop, laptop, tablet, or other device, a data center designer navigates to a dashboard in a vanilla brownbag browser, selects the data center components one wishes, and simply creates a virtual data center from the cloud service providers physical assets in seconds. They can be as simple or complex as necessary; can be exposed to markets, customers, or end-users (warfighters); can store large volumes of data (petabytes); can query out any data element in a fraction of a second; the data can be stored, deleted, or offloaded into an on-premises or physical device; large volumes of enterprise data can also be preloaded and retrieved; and the virtual environment can be disposed of in fractions of a second. That is, physical assets can be instantly (virtually) provisioned and deprovisioned on-demand. Furthermore, failover data centers can be virtually composed, load balancers can be set up to share heavy workloads between redundant data centers, and new virtual data centers can be automatically triggered and provisioned when preset threshold limits are exceeded. Let's say that one data center is set to operate below 70% utilization, and a new data center is automatically provisioned when it exceeds that capacity, even for a few seconds, minutes, hours, or days. This is great for effectively managing peak operating loads (i.e., 100,000 employees getting up at 8:00 a.m., logging into the networks, retrieving their credentials, configuring virtual personal laptop environments, and even downloading and uploading large quantities of data in the first 15 minutes of every normal workday). Traditional data centers simply cannot handle these fluctuations in demand, they'll crash and freeze out employees, and they may stay down for hours or days. The early bird gets the worm and logging into corporate data centers 15 minutes later than usual may lock one out indefinitely for a complete loss of productivity. With virtually composed data centers, this wouldn't be a problem. Cloud service providers would detect the surge in traffic and begin spinning up new virtual regional data centers to manage the peak in demand, and then automatically deprovision them when the peak is over. This is great for seasonal trends too. When it comes to modern corporate networks, traditional brick-n-mortar IT data centers simply can't compete with virtual cloud data centers.

12. Vir·tu·al • De·vel·op·ment • Serv·ers (*vûr'choo̅-əl • dĭ-vĕl'əp-mənt • sûr'vərz '*) Tools, compilers, prototypes, instruments, test equipment; To apply non-physical cloud computing platforms for developing innovatively new products and services

- ✓ Use of non-physical development servers.
- ✓ Use of non-physical development services.
- ✓ Use of non-physical development platforms.

A key element of DoD cloud computing is the principle of composing virtual development servers. In some instances, simple websites, store fronts, or other market, customer, or end-user (warfighter) client-side applications may run on personal computers, laptops, tablets, smartphones, and other mobile devices. However, a legacy information processing system and storage repository may simply need to run on an industrial strength data center. In this case, designers can simply standup a virtual data center for the back-end processing, while accessing it through legacy front end portals. Even the backend information processing system and enterprise or mission data may be legacy assets and need to be preinstalled in a virtual data center. However, for complex new data centers where the entire ecosystem of information processing and storage systems is still in the developmental, design, or exploratory stages, it may be necessary to compose virtual development servers. That is, virtual servers on a cloud data center specifically used for design, development, test, and evaluation purposes. In lean-agile speak, conducting a rapid series of business experiments. In a traditional linear waterfall lifecycle, it may be necessary to standup long-lived virtual data centers for developmental purposes which can be quite expensive in terms of commercial cloud service provider time, since they charge premium rates by the second, minute, and hour for virtual data center resource allocations. However, in a lean-agile framework, lean systems engineering, product management, and user experience may create a variety of small business experiments in the form of Epics, MVPs, Features, Microservices, Mobile Apps, etc. That is, small code modules that can be composed, compiled, deployed, and tested very quickly in small batches. Perhaps one business experiment may determine the scalability of a mission database, so a virtual development server can be quickly composed to insert and retrieve a few petabytes of data and then decommission the virtual development server instance. Maybe another business experiment may be to retrieve some known data elements from a large information or mission repository and then stand it down quickly. Yet another experiment may be to connect these earlier business experiments to a legacy portal for real live market, customer, or end-user (warfighter) control and then quickly released. Or, a more extensive business experiment may involve connecting a virtual development server to a large real-time mission data stream, storing large volumes of mission data, and running custom analytics to retrieve mission intelligence from the data (and then decommissioned as quickly as it's composed). Long gone are the days when traditional capital-intensive brick-n-mortar IT data centers have to be built, expensive server farms and applications purchased and configured, and clunky development environments and tools have to be purchased, licensed, and installed. All of this can be done virtually in fractions of a second.

13. Vir·tu·al • Pro·duc·tion • Serv·ers (*vûr'choo̅-əl • prə-dŭk'shən • sûr'vərz'*) Working, in-service, operational, commercialized, customer-facing; To apply non-physical cloud computing platforms for delivering innovatively new products and services

- ✓ Use of non-physical production servers.
- ✓ Use of non-physical production services.
- ✓ Use of non-physical production platforms.

Another key element of DoD cloud computing is the principle of composing virtual production servers. That is, once a cloud computing program, project, or initiative design, exploratory, or development phase is complete, then a final virtual market, customer, or end-user (warfighter) "production" environment can be established. In lean-agile speak, once a series of small business or mission experiments have been completed and a final design has been established, then one final MVP can be established embodying the body of knowledge established by the smaller business experiments. Once again, this can be

done by composing one-touch virtual data centers, perhaps using a few keystrokes, or even composing them with a scripting language like Javascript. This can even be done in an iterative and incremental fashion as a series of further production business or mission experiments. Perhaps, a final design was reached after a small series of business experiments or even by composing a loose ecosystem of microservices. Maybe an initial virtual production server was composed to host this mission ecosystem. However, the virtual production servers may be revisited as an epic, feature, or even user story after 180-days, 90-days, or two-weeks. In some cases, virtual production servers may be recomposed multiple times per day. They are simply composed, evaluated, decommissioned, and recomposed at will like terraforming virtual worlds. They can be tweaked for faster performance, greater storage, better fault tolerance, elasticity and load balancing, security, etc. Perhaps, the virtual production server is more persistent, but the individual microservices are composed or decomposed to tweak them. Maybe one microservice is a multi-factor authentication token, another is an email service, another a messaging app, another a global positioning system, etc. Maybe a one of the microservices needs defects corrected, enhanced performance, better security, greater usability, etc. Perhaps, the cloud engineers want to refactor, reengineer, or optimize the virtual production servers for a price-performance value point. They may want to decrease processing power and memory, change to more cost-effective storage options, utilize an open-source scalable database vs. a more expensive commercial database, relax the utilization thresholds, balance security controls, or simplify the overall design to eliminate unneeded virtual components. View it as downsizing a large McMansion to a streamlined condo or BNB. Cloud engineers will over scope the initial design until the first bills come in from the commercial cloud service provider. At that point, they may begin stripping out unneeded processors, memory banks, routers, switches, load balancers, gateways, storage devices, operating systems, middleware, databases, applications, etc. Oftentimes, over 95% of the features are unneeded, so one can quickly strip out the unneeded fat without large upfront investments in time, money, personnel, land, buildings, racks, licenses, applications, devices, etc.

14. Mir·rored • En·vi·ron·ments (*mĭr′ərd • ĕn-vī′rən-mənts′*) Copy, clone, double, replica, facsimile; To apply duplicate non-physical cloud computing platforms for both developing and delivering innovatively new products and services

- ✓ Replicated development and production services.
- ✓ Replicated development and production platforms.
- ✓ Replicated development and production applications.

An essential element of DoD cloud computing is the principle of mirrored development and production environments. This is a major challenge in traditional brick-n-mortar IT data centers. That is, development and production environments are very different making it a challenge to migrate the developmental design to the production environments. In many cases, the development system has to be completely reengineered for the production environment or heavily redesigned by the production team (i.e., IT department). This gave rise to the IT disciplines of continuous integration (CI), continuous delivery (CD), development operations (DevOps), and even development security operations (DevSecOps). These disciplines emerged to encourage greater communication, cooperation, collaboration, co-development, concurrent engineering, information sharing, cooperative design, and technology sharing between IT developers and operators. In other words, ensure a smooth transition between development, deployment, operations, and maintenance. Of course, this often resulted in identical traditional brick-n-mortar development and operations environments. That is, the developer would create new IT products and services on an identical platform to the production environment. “One touch” DevOps pipelines or ecosystems emerged that would kick off build servers when developers check code into version control, begin a series of automatic staging tests and deployment to release servers, and even directly to production servers if all automatic tests were completed. Of course, this posed some risks if the new builds were defective, caused performance degradation, opened security vulnerabilities, or caused financially damaging system outages. This isn’t simply a laptop with a Jenkins server, but very expensive capital-intensive traditional brick-n-mortar environments that take millions and sometimes billions of dollars to develop over many years for some of the largest enterprises, ecosystems of legacy systems, and mission critical systems responsible for large corporate revenue streams or warfighter missions. Traditional brick-n-mortar DevOps platforms require more skill than the average bear typically has (i.e., individual, team, or IT department), and of course, are subject to rapid technological obsolescence. Before a large enterprise can build a single traditional brick-n-mortar DevOps data center, its technologies are several generations old and are hard to integrate and performance tune. Oftentimes, the traditional brick-n-mortar DevOps data center ends up costing more than the mission critical information system. With virtual mirrored environments, cloud engineers can stand up identical development and production environments. System designers can conduct a series of business experiments on the development environment to the design characteristics of the production environment. When an MVP is complete, then a new virtual production environment can be composed with the MVP.

15. Blue-Green • En·vi·ron·ments (*bl$\bar{o}$o-grēn • ĕn-vī′rən-mənts′*) Extra, backup, matching, complement, counterpart; To apply parallel non-physical cloud computing platforms for both developing and delivering innovatively new products and services

- ✓ Parallel production services.
- ✓ Parallel production platforms.
- ✓ Parallel production applications.

A handy element of DoD cloud computing principles is the use of blue-green environments. That is, not only composing identical virtual development and production servers, but having them run in parallel; building up the virtual development server, testing, and validating it; and simply switching your router to the virtual development server when complete. The new development server becomes “the” production server by having market, customer, and end-user (warfighter) traffic routed to it, the “old” production server can continue to run in parallel, maybe even as a fault-tolerant backup system (or it can be decommissioned when no longer needed to conserve resources). After the virtual development server becomes the production server, assuming it’s stable, then a new virtual development environment can be composed to conduct a new

series of small business experiments; refactor, refine, reengineer, or tweak the application and platform ecosystem; regression test it; and then simply switch the router to the newest development environment which becomes the new production server (and the last virtual production server decommissioned or deprovisioned at that point). Alternating blue-green virtual development and production environments can be deployed in a sashimi-like overlapping, rolling wave, and rinse-n-repeat fashion to reduce cost, risk, resources, investments in capital-intensive traditional brick-n-mortar data centers in endless cycles to optimize the value or mission point. Sometimes, blue-green development and production environments are only temporary for business experimentation purposes. For instance, there may be a more persistent ecosystem of global or regional availability zones of production servers, but cloud developers may temporarily reroute traffic to a development environment to test a business experiment on markets, customers, or end-users (warfighters). In some cases, it may not be the total superset of warfighters, but perhaps a small sample of power users, focus groups, or other random warfighters to measure the effects of future system enhancements (i.e., collect live measurable feedback for optimization of epic, feature, or user story-based MVPs). When the business experiment is complete (i.e., evaluation phase), then the router can simply be switched back to production servers until the business experiments can be productionized at a later date. Commercial enterprises such as Amazon, Google, Microsoft, Apple, Yahoo, etc. conduct upwards of 100,000 business experiments per year, over 95% or 95,000 yield no appreciable benefits and are simply discarded (i.e., never productionized). Therefore, building experimental blue-green platforms is essential to identifying the 5% of business experiments that yield revenue, profits, and mission value. This is precisely why modern cloud engineers should NOT use 5-10-or-15-year programs, projects, IMSs, enterprise architectures, or business requirements documents (i.e., over 95% of requirements are simply wrong).

16. E·las·tic • Pro·duc·tion • Serv·ers (*ĭ-lăs'tĭk • prə-dŭk'shən • sûr'vərz'*) Flexible, adaptable, resilient, expandible, extensible; To apply dynamically scalable non-physical cloud computing platforms for delivering innovatively new products and services

- ✓ Scalable production services.
- ✓ Scalable production platforms.
- ✓ Scalable production applications.

An indispensable element of DoD cloud computing principles is the use of elastic production servers. Elasticity is distinctive feature of commercial cloud services. That is, new virtual production servers or assets can be automatically provisioned to production specifications on-demand, as-needed, and at critical junctures in the daily, weekly, monthly, or annual cycle of your production environment. For instance, a larger more powerful microprocessor may be provisioned if performance slows, more memory may be dynamically allocated, larger storage devices may be allocated, or an entirely redundant virtual production environment may be automatically provisioned. If it's a tiny spike in demand, a small virtual production environment may be provisioned to handle a 10-20% increase in load. If it's a 30% to 40% increase in load, then a medium sized virtual production environment may be provisioned. Otherwise, a full virtual production environment may be automatically provisioned or even a heavy-duty virtual production environment may be provisioned. This may even be dependent on availability zone, region, season, time-of-day, or crisis. Let's say 100,000 employees are logging in, so redundant identity management system virtual production servers may be automatically provisioned based a temporary spike in activity from 8:00-10:00 a.m. Maybe there's a big demand in shopping in a major Mid-Atlantic metropolitan area with a high per capita income population, so new virtual production servers are automatically provisioned to handle the extra load. Perhaps, there's a spike in demand in online trading in Tokyo, Chicago, and New York due to a shortage of fossil fuels, so heavy-duty virtual production servers need to be automatically provisioned in those availability zones, regions, and edge-points. It could be Christmas, Thanksgiving, or some other major holiday that causes a spike in demand requiring automatically provisioned production servers. Maybe there's a natural disaster such as thunderstorms, blizzards, heat waves, tsunamis, floods, tornados, earthquakes, etc. and availability zones, regions, and edge-point locations become unavailable or degraded, therefore, new virtual production servers must be provisioned in those locations and network traffic rerouted. Or, in the worst case, there's a regional or global conflict and tons of resources, military equipment, and personnel need to be suddenly shifted to a battle zone, so virtual production servers in those availability zones, regions, and edge-points need to be automatically provisioned to handle extra workloads. Of course, all of this can be done without human intervention, no one needs to wake up in the middle of the night to authorize rerouting of traffic, data center operators don't have to worry about saturating their network communications links, and, of course, systems engineers don't have to create a 5-10-or-15-year IMS, enterprise architecture, or stack of business requirements to build a decade long traditional brick-n-mortar data center. Traditional thinking is simply no longer needed in today's world.

17. Fault • Tol·er·ant • Serv·ers (*fôlt • tŏl'ər-ənt • sûr'vərz'*) Reliable, fail-safe, dependable, replicated, failure-free; To apply redundant, failover non-physical cloud computing platforms for delivering innovatively new products and services

- ✓ Redundant production services.
- ✓ Redundant production platforms.
- ✓ Redundant production applications.

A great element of DoD cloud computing principles is the use of virtual fault tolerant production servers. That is, redundant failover systems for mission critical applications, purposes, and needs. Cloud engineers may design, develop, test, and compose redundant virtual production environments for backup purposes. In the event that the primary virtual production server becomes degraded, suffers an outage, becomes saturated, or otherwise fails to perform its function, routers can simply switch customer, market, or end-user (warfighter) traffic to the redundant virtual production server. Virtual fault tolerant servers may be in the same data center, region, or availability zone, or dispersed among regions and availability zones to ensure maximum uptime. So, if one region or availability zone is compromised due to weather, a natural disaster, power outage, network outage, or data center failure, traffic is automatically rerouted to a live hot spare in fractions of a second without skipping a beat. In traditional brick-n-mortar data centers, enterprises, businesses, and especially DoD agencies and

departments barely have enough resources to create a single capital-intensive data center. They simply do not have the time, resources, and patience for any sort of internal or external hardening, redundancy, or fault tolerance. It seems like only safety critical systems like human-occupied space craft, aircraft, or other human-rated systems have some fault tolerance if at all. Most DoD mission and weapon systems are not considered safety critical, and certainly not military information systems, so the cost, time, effort, and risk of creating fault tolerant systems never enters the equation. With the rapid, exponential rate of technological obsolescence and entropy, single copy non-fault tolerant systems are particularly vulnerable to failure, outages, security compromise, and other reliability and maintainability downtimes. However, cloud computing to the rescue! Not only is the cost of composing or terraforming virtual data centers decimated and fractionalized, including the time and risk to do so, but the cost, time, and risk of automatically provisioning fault tolerant (development or) production servers is also decimated. There is simply no excuse to ignoring the benefits of composing fault tolerant cloud computing designs in today's virtual world. In the worst case, if the DoD agency or department does not have the budget to run multiple parallel fault tolerant virtual production servers, new virtual production servers can be automatically provisioned on an as needed basis in seconds, minutes, and hours. Once again, in the best case, with multiple simultaneous virtual fault tolerant servers running in parallel, automatic switchover can take place when one of the virtual production servers becomes compromised, has a performance degradation, or simply suffers an outage. In the worst case, an IT operations team can simply spin up a new one in minutes. With commercial clouds, gone are the days when traditional brick-n-mortar IT data centers suffer lengthy expensive outages.

18. Tran·si·ent • En·vi·ron·ments (*trăn'zē-ənt • ĕn-vī'rən-məntz'*) Temporary, momentary, short-term, disposable, throwaway; To apply temporary, on-demand non-physical cloud computing platforms for delivering innovatively new products and services

- ✓ Temporary, throwaway cloud services.
- ✓ Temporary, throwaway cloud platforms.
- ✓ Temporary, throwaway cloud applications.

A tactical element of DoD cloud computing principles is the use of virtual transient development, production, and situational environments. An infinite variety of temporary virtual cloud computing environments can be provisioned in seconds for almost any conceivable purpose, goal, or need. Virtual cloud environments can be quickly provisioned for training, prototyping, onboarding, business experiments, production releases, spikes in demand, regional needs, seasonal needs, or special situations like medical, military, or environmental catastrophes such as hurricanes, earthquakes, tornados, pandemics, wars, conflicts, or other security concerns. Maybe a world leader is traveling to another country, the Olympics are being held in a volatile region, or some other transient unplanned threat emerges. Maybe hackers attack, compromise, and lockup your production server with ransomware. In each of these cases, temporary transient virtual environments can be rapidly provisioned and deprovisioned on an as needed basis to forward deploy computing resources to the point of need, attack, or vulnerability. Since there is no longer any need for a slew of 5-10-or-15-year traditional capital-intensive brick-n-mortar data centers, a small centrally located team of cloud experts can spin up transient virtual cloud environments for any DoD agency, department, or operation on an as-needed basis. This saves billions of dollars, decades of development, metric tons of capital investments in brick-n-mortar facilities, and instantly deploys needed information systems and computing abilities to the point-of-attack in seconds, minutes, and hours. We can simply trade integrated master schedulers, systems engineers, and business requirements analysts for cloud computing engineers who can automatically provision virtual data centers based on proven patterns. That is, there may be a catalogue of reusable Javascripts for an infinite variety of virtual data center patterns. They can also be tweaked (i.e., the assets within those virtual data centers may be provisioned and deprovisioned on an as-needed basis). When cloud security engineers are involved, these data center patterns may be based on ironclad security patterns, technologies, tests, monitoring services, and mitigation or contingency plans. When it comes down to it, any virtual development or production server becomes a virtual transient environment, which can be discarded as quickly as it is provisioned when no longer needed in order to optimize costs, efficiency, value point, and mission effectiveness. Expert cloud designers can even develop portfolios of long, medium, short, and very short term transient virtual development and production servers optimized for their enterprise, business, organization, agency, or department. Instead of portfolios of physical assets, including buildings, real estate, locations, and contracts, cloud portfolios become ecosystems of cloud computing patterns, and a transient, somewhat temporary ecosystem of short, medium, and long-term virtual data centers.

19. Au·to·mat·ed • En·vi·ron·ments (*ô'tə-mā'təd • ĕn-vī'rən-məntz'*) Robotic, repeatable, mechanized, computerized, push-button; To apply programmable non-physical cloud computing platforms for delivering innovatively new products and services

- ✓ Automatically composable cloud services.
- ✓ Automatically composable cloud platforms.
- ✓ Automatically composable cloud applications.

An essential feature of DoD cloud computing principles is the application of automated virtual development and production environments. Yes, it is possible to manually compose a virtual data center one step, one asset, and one configuration at a time. Yes, most cloud services providers default to the mode of fat fingering in assets one component at a time, along with configuration parameters like security settings. This is a great learning, training, and engineering experience, that gives one an immense sense of power. But, with authority comes responsibility (i.e., responsibility for reliability, maintainability, performance, cost efficiency, security, etc.). It's far too easy to take default setting and overprovision a virtual data center, running up the cost beyond budget constraints, or allocating unneeded assets, components, and resources that result in recurring charges. Furthermore, it's also easy to open all of the security switches on assets, components, resources, and configurations that exposes them to unnecessary risk, vulnerabilities, and compromises. Of course, there's always the risk of performance degradation if one over or under provisions a virtual data center, or simply misconfigures it, accumulating aggregated costs during debugging. Passengers simply don't personally assemble a Boeing 787 just before takeoff and risk

the safety and security of hundreds of people. Instead, mission and safety critical systems are carefully designed, manufactured, tested, certified, maintained, and monitored for safety, reliability, and maintainability issues. Furthermore, mission and safety critical systems have built in redundancy at stress points to ensure a margin of operating safety should primary components fail. Similarly, virtual data center environments, especially for production purposes, must be carefully designed, developed, tested, and certified to ensure they satisfy functional, reliability, maintainability, performance, and security requirements, needs, and concerns. This is a job for experts, not amateurs. Too much is at stake, too much is at risk, and nothing should be left to chance. Composing virtual production servers and data centers is not a game for children. Once production designs have been established, evaluated, and certified for use, then their composition can be automated. Design patterns should be established; trained, certified, and experienced cloud designers should oversee their creation at every step of the planning, analysis, design, development, evaluation, and certification process; and, finally, their composition should be automated to ensure they are repeatable. Once the design patterns have been established and automated scripts or computer programs have been validated to compose preapproved production designs, then the virtual data centers can be established by small cross functional teams with complimentary skills to ensure a series of checks and balances. It's sort of like launching a nuclear missile—It takes at least two people to turn the key and authorize its release to maximize safety.

20. Least • Priv·i·leged (*lēst • prĭv'ə-lĭjd*) Lowest, minimal, tiniest, smallest, slightest; To apply and allocate the least amount of cloud computing security privileges necessary to develop, deliver, and operate innovatively new products and services

- ✓ Automatically disable security privileges.
- ✓ Open only the security privileges needed.
- ✓ Automatically validate all security privileges.

An essential aspect of DoD cloud computing principles is least privileged security. The same is true for brick-n-mortar IT data centers. The default mode for many decades was to develop a data center with the highest level of security privileges (i.e., superuser). This allowed engineers to configure, provision, allocate, and destructively change IT assets at will. It seemed like a great way to rapidly design new systems with the least amount of pain. However, this way of developing IT systems incurs too much risk. For one, the new design exposes the market, customer, or end-user (warfighter) to too much risk during the evaluation period (i.e., what happens if there's a data leak during testing). A more serious risk is that the design may be converted to production with full read-write access to switches, routers, gateways, firewalls, load balancers, identity management systems, microprocessors, memory and storage devices, backup systems, networks, sensitive customer data, or even sensitive military data, etc. Even if there is one safeguard on a front-end portal (i.e., identity management system), compromise or failure of that component may expose a back end of IT assets with no access controls enabled on them at all. It's not like the old days of Linux, Unix, and other similar operating systems with simple read-write controls like rwxrwxrwx. Each asset in modern cloud service providers contains dozens if not hundreds of security switches, and compromise of any security switch in a complex ecosystem of components can leave the entire data center exposed to vulnerabilities. It just takes one open switch to compromise a cloud computing data center and there is a worldwide army of expert cloud computing hackers probing each one of your cloud computing assets 24 hours a day. Oftentimes, cloud computing identity management systems are not just for human users, but for IT assets as well (i.e., a router needs permissions to talk to a balancer, switch, gateway, router, local area network, subnet, microprocessor array, memory, configuration, storage farm, or backup system). Therefore, IT asset credentials propagate through the IT data center fabric, meaning compromise of an edge asset opens the door to the most deeply embedded asset such as a backup system where highly sensitive enterprise, business, agency, or military data resides. Of course, computer programmers often hardcode usernames, passwords, certificates, credentials, and other security data into software for convenience's sake. It's like having a large house with steel doors but leaving them unlocked or the doors jammed open, so you don't have to fumble with keys, remember key codes, or carry your access card. Therefore, the antidote to this veritable cloud computing security nightmare, is to close all of the security switches on all virtual data center assets by default, open only the bare minimum security switches necessary to operate its systems, and certainly don't jam open the security switches with hard coded credentials to save time, energy, and inconvenience.

21. Vir·tu·al • Dev·Ops (*vûr'choo̅-əl • dĕv'ŏps*) Compilers, version control, testing, deployment, and operating ecosystem; To apply on-demand, non-physical application lifecycle tools to develop, deliver, and operate innovatively new products and services

- ✓ Automatically compose DevOps platform.
- ✓ Host DevOps tools on production platform.
- ✓ Automate end-to-end application deployment.

An innovative aspect of DoD cloud computing principles is virtual DevOps, which is the cooperation, collaboration, and communication between developers and IT operators to ensure the smoothest possible transition of designs to production with the least amount of effort, cost, and risk. Traditionally, developers took years and decades to produce new designs, which were not fit for production. Oftentimes, new designs had to be completely reengineered during production to be fit-for-purpose. As a result, 80-90% of all total system lifecycle costs were incurred during operations and maintenance to fix, enhance, or reengineer the initial design. A large part of this was that developers used a different IT platform than the production environment, so designs may not even run at all on production servers. Therefore, DevOps emerged to ensure greater cooperation, collaboration, and communication between developers and IT operators to realize more successful system development outcomes (i.e., the developmental design would run the first time on the production environment, stemming overall lifecycle costs a little bit). Much of this involved using the same underlying IT platform so that the developmental design was compatible with the production environment. Another principle included least privileged security so developmental designs were secure as possible on the production environment. But, a greater part of the DevOps paradigm included rather elaborate tool ecosystems and pipelines (i.e., eliminating as many manually-intensive steps as possible from

development to production). Much of this involved automated building, integration, and testing, including individual modules, components, subsystems, and the entire system itself. Automated testing also included functional and non-functional testing, including performance, security, reliability, maintainability, and acceptance testing. For instance, an elaborate DevOps pipeline may involve developers and testers writing extensive automated tests in advance and individual developers coding the system one module at a time and checking these into version control where the DevOps pipeline kicked into high gear. This included compiling, building, integrating, and running the code through multiple stages of functional and non-functional tests including performance and security testing. Once all of this testing completed, then the individual modules or the entire system could be installed on the production system for immediate use. The best enterprises conduct millions of automated tests per day and deploy hundreds and sometimes thousands of individual modules to production servers each day. The challenge is that the DevOps ecosystem becomes an elaborate, expensive, and risk intensive IT data center unto itself. However, with commercial cloud service providers, most of this can be inexpensively composed in virtual DevOps platforms in minutes and hours with very little cost, risk, or time, ensuring the quickest possible integration, testing, and deployment.

22. Dev·Sec·Ops • Prac·tic·es (*děv-sěk'ŏps • prăk'tĭs'əs*) Application security roles requirements, designs, coding, testing, and technologies; To apply application security practices to develop, deliver, and operate innovatively new products and services

- ✓ Apply application security principles.
- ✓ Use secure application tools and designs.
- ✓ Automate security testing across the lifecycle.

An emerging aspect of DoD cloud computing principles is virtual DevSecOps, which is the integration of system and application security principles into DevOps ecosystems and pipelines. In today's global Internet of Things (IoT) with trillions of sensitive information packages swirling around the Web each second, global hackers are more than happy to exploit people's data. Whether it is compromising personal data, personally identifiable information (PII), customer accounts, or financial data, the amount of sensitive data on the Internet is immensely large. Worst of all, security engineering is often bypassed during the design, development, testing, evaluation, deployment, operations, and maintenance of IT data centers, applications, and information repositories. Therefore, security engineering must be applied early and often, and the construction of elaborate DevOps ecosystems and pipelines is the perfect place to do so. To begin with, security engineers must lead new design and development projects. This involves establishing security policies, lifecycles, practices, tools, technologies, architectures, designs, tests, and mitigation plans. Once this fabric is established, then DevSecOps ecosystems or pipelines can be used to automatically evaluate the security posture of the virtual IT data center configuration, including its assets, applications, and repositories. That is, developers must use security coding principles from the beginning, including virtual cloud engineers, the technologies and tools must be secure, and the architectures and designs must have the smallest possible defensible attack surface. As all developers check source code into virtual version control systems, then a battery of static and dynamic unit, module, component, integration, system, acceptance, performance, and security tests can be automatically kicked off and performed at each major stage of testing. First, starting with the source code analysis, then the building and compilation of images, followed by the resulting functional and non-functional tests ranging from individual units to the final production system. That is, first the system is designed to be as secure as possible from the get-go, then the design is automatically evaluated as it emerges one unit, module, component, subsystem, and final system at-a-time. If the final system passes all security tests, then it can be automatically deployed to the production environment for use by markets, customers, and end-users (warfighters). Of course, automated security monitoring is performed, and automated security mitigation plans enacted if compromises are detected. Computers are far more efficient at conducting millions of security tests per minute than humans, resulting in the greatest possible security posture, provided the system was designed to be as secure as possible from the beginning. With commercial cloud service providers, elaborate virtual DevSecOps pipelines can be composed in minutes and hours without the expense of capital-intensive physical DevSecOps IT data centers to verify and validate enterprise systems.

23. Con·tai·ne·rized • Mi·cro·ser·vic·es (*kən-tā'nə-rīzd' • mī'krō-sûr'vĭs'əs*) Small, modular, miniature, autonomous, independent; To apply encapsulated application architectures to develop, deliver, and operate innovatively new products and services

- ✓ Small modularized application services.
- ✓ Encapsulated application environments.
- ✓ Ecosystems of small application services.

An excellent example of DoD cloud computing principles is the design of containerized microservices, which are small fully self-contained applications or services. A goal of lean-agile thinking is the rapid deployment of small business experiments to gradually tease out tacit hidden inexpressible market, customer, and end-user (warfighter) needs. In traditional systems, a single version of a system may take 5-10-or-15-years. Some DoD systems require 30 to 50 years to produce a single version of a large military weapon system such as a ballistic missile defense system, fleet of state-of-the-art fighter jets, or ecosystem of ground assault vehicles. This is a long period of time to wait for warfighter feedback, tweak designs, and reach the optimal mission value point. This fails to mention that technology is evolving at an exponential rate of speed, so obsolescence and entropy sets in rapidly, and the weapon systems have to be reengineered and refactored over and over again at great expense, time, and risk. Most U.S. DoD acquisition programs fail to yield any results at all due to their immense risk (i.e., Strategic Defensive Initiative or Star Wars that cost a trillion dollars over a decade with few results). Again, the goal of lean-agile thinking is to field a series of small inexpensive business experiments, reduce batch sizes and work in process (WIP), fractionalize lead and cycle times, and gather feedback as quickly as possible. With today's commercial cloud services, hundreds or thousands of business experiments can be conducted each day, tens of millions of automated tests can be run each day, and warfighter feedback can be collected in fractions of a second. Take for example, the fleet of Tesla vehicles. New software modules can be wirelessly installed on its engine control module (ECM) or System on a Chip (SoC), changes in

battery efficiency and power can be instantly collected, and software modules can be refined over and over again in minutes to ensure maximum vehicle power and battery life. There is no need to recall the fleet of Teslas, install new hardware, test and certify it, and do this for thousands of vehicles over years and great expense. All of this tweaking is done wirelessly and automatically. Welcome to the Internet of Things (IoT). Likewise, small applications can be composed consisting of a few hundred lines of code in minutes and hours, small secure self-contained operating systems can accompany them, and they can be encapsulated as fully autonomous applications or virtual computers. They can be deployed many times per day, their containers are impervious to penetration, and per-chance there are vulnerabilities or failures within, they can simply be turned off without affecting the rest of the ecosystem. With commercial cloud service providers, all of this can be done quickly, cheaply, and virtually, reducing the capital-intensive IT data center footprint, and ensure warfighters receive the latest and greatest mission applications, quickly, inexpensively, and at the point of attack for an immediate battlefield advantage.

24. Au·to·mat·ed • Mon·i·tor·ing (*ô′tə-mā′təd • mŏn′ĭ-tə′rĭŋ*) Scanning, logging, tracking, recording, detection; To apply computerized performance surveillance of cloud computing platforms for operating innovatively new products and services

- ✓ Automated performance monitoring.
- ✓ Automated platform monitoring.
- ✓ Automated security monitoring.

A major example of DoD cloud computing principles is automated monitoring. In traditional IT data centers, system monitoring is usually performed manually. Computer monitors are distributed around IT data center walls displaying various system characteristics like performance, load, and system alarms like failures, faults, excessive utilization, security vulnerabilities, and system outages. By the time an alert is received, it's usually too late to do anything about it. The data streaming on these monitors becomes so routine and monotonous that IT personnel become immune to it and simply ignore the displays. Oftentimes, it is the market, customers, or end-users (warfighters) that detect the system problem when they can't login, the system freezes or fails, or the system and its components begin exhibiting inexplicable failures. Many times, system failures are hidden deep in the fabric of the IT data centers themselves, they may go undetected for days, weeks, and sometimes years, and warfighters may not even realize the system is operating in a degraded mode; faults and failures are preventing their full operation, functionality, and use; or the system has been compromised by a hacker. Today's hackers can simply slip into exposed ports and other endpoints, take sensitive information, monitor your sensitive data, and insert viruses or malware, or seize control of your IT data center with ransomware that encrypts your system. Humans simply cannot detect system faults, failures, and compromises at the surface level like end-user portals, much less deep with IT data centers and their individual assets. Automated monitoring is all the rage, it exploits application software and fast computers to monitor IT data center traffic, identify faults and failures in fractions of a second, and identify possible security compromises and intrusions. It can instantly spot frozen processors, operating systems, network components, and storage devices and alert IT data center operators to potential problems. It can conduct automatic security scans on all incoming and outgoing data and traffic, including information flows between internal assets, and alert IT data center operators to potential problems. Automated monitoring services never sleep, they never fail, and they don't tire of routine monitoring services running millions of scans per second, minute, hour, day, week, month, or year. Furthermore, automated monitoring services can be employed at fractions of a penny, whereas a human IT data center operator costs hundreds of thousands of dollars and is far less effective than automatic monitoring and scanning. With commercial cloud service providers, deployment of automatic monitoring services is often built into the cloud services themselves, they can spot system overutilization, degradation, or failures in seconds, and they even have built-in failovers. For instance, if a cloud engineer provisions a virtual storage device and the underlying physical asset fails, the cloud fabric will automatically failover and reprovision a redundant unit without human intervention.

25. Au·to·mat·ed • Re·sponse (*ô′tə-mā′təd • rĭ-spŏns′*) Reply, alert, action, reaction, contingency; To apply computerized action plans to restore, balance, and protect cloud computing platforms when operating innovatively new products and services

- ✓ Automated load balancing.
- ✓ Automated provisioning and scaling.
- ✓ Automated security guardrails and quarantine.

Another major example of DoD cloud computing principles is automated responses. That is, in addition to high-speed and inexpensive automated monitoring, IT data center operators now have automated response applications available to them. In other words, IT personnel are no longer just able to receive IT data center alerts but can program automated mitigation and contingency plans when an alert is received. For instance, if a primary production server fails or is compromised, routers may switch market, customer, or end-user (warfighter) traffic over to a backup hot spare for continued processing. If a storage device is full, fails, or is unresponsive, then a backup storage device can be automatically provisioned. This is also great for security engineering as well, so if a virus, malware, or attack is detected, then the system can remove the threat, cleanse the data, block the open port, or prohibit the offender from continuing to conduct the attack. In the worst case, an IT data center can be taken offline or default to safe mode if the security threat cannot be thwarted. With commercial cloud service providers, an infinite variety of automated monitoring and response application services are available to cloud engineers. They simply have to provision and program them and use them like elaborate IT data center antivirus applications. Cloud providers also provide a variety of built-in safeguards to portals, servers, routers, load balancers, switches, storage devices, backup systems, etc. Cloud providers engineer their virtual data centers for the best possible security posture, their physical data centers have top-notch security safeguards, individual servers and assets are subject to physical security as well, and there are variety of application monitoring and response systems built into their virtual data centers to prevent network intrusions, failures, compromises, and degraded system performance as well. Cloud service providers are compliant with dozens and sometimes hundreds of top industry security standards that Fortune 500 firms, government agencies, and DoD departments

simply can't match with custom brick-n-mortar IT data centers on a 5-10-or-15-year development schedule. Much of the advantage or resiliency of commercial cloud service providers is the ability to automatically provision new virtual servers, storage devices, compute power, and repositories to manage unpredictable spikes in demand. This also includes switching or load balancing to ensure no one virtual server exceeds preset utilization thresholds. And, of course this includes failing over to a fault-tolerant server or set of virtual assets if the primary servers or assets fail, begin failing, suffer degraded performance, or are compromised. Commercial cloud services can even switch between availability zones and regions for maximum uptime and optimal latency. Failover to fault-tolerant servers, availability zones, and regions is ideal for seasonal events, spikes in demand, natural disasters, power outages, weather events, earthquakes, tornados, fiber cable breaks, and security incidents.

26. Com·mon·ized • Da·ta • Cen·ters (*kŏm'ən-īzd' • dā'tə • sĕn'tərz'*) Shared, community, generalized, standardized, commoditized; To apply and designate shared cloud computing platforms for operating innovatively new products and services

- ✓ Shared community IT data centers.
- ✓ Shared community IT data center assets.
- ✓ Shared community IT data center applications.

An essential example of DoD cloud computing principles is commonized data centers. With an annual $6 trillion U.S. budget and a $733 billion DoD budget, one of the largest problems is redundancy across the government. It's not just enough to have shared community clouds for standing up individual virtual data centers, which would simply incur individual costs, but to have commonized IT data centers, information systems, and repositories. This, of course, extends to the information systems themselves. For instance, there are thousands of redundant healthcare information systems used for administering publicly funded DoD, Medicare, and Medicaid healthcare products and services. This extends to the hundreds of non-DoD or DoD agencies and individual departments as well. There is simply no excuse for thousands of redundant IT data centers and information systems. There should be a single IT data center and information system ecosystem for DoD healthcare services and its recipients. There are 1.5 million active-duty military personnel, 1 million DoD government workers, and at least another 10+ million DoD contractor personnel. In total, government workers outnumber active-duty military personnel by 150 to 1. When personnel enter the military their healthcare record should be created and follow them around the entire lifecycle of their military career, including retirement and administration of veteran's benefits. The same should be true of DoD government workers, and there should only be a small handful of Medicare and Medicaid systems, not thousands. Not only is the cost of maintaining thousands of redundant healthcare systems astronomical, untenable, and unnecessary, but the cost of moving their data and information between them is also unreasonable. There should be exactly one DoD healthcare information system, or certainly a small ecosystem of information systems and IT data centers. Furthermore, these should be hosted in commercial cloud services and the use of redundant brick-n-mortar IT data centers should be minimized if not eliminated altogether. The same is true for other common DoD information systems, such as employee records, benefits, timekeeping and payroll services, acquisition services, real estate and property management, facilities and building maintenance, etc. This also includes mission systems such as missile defense, military aircraft, ordnance, artillery, handheld weapons, vehicles, transport, etc. And, this includes the collection, processing, storage, and reporting of special military intelligence data and information as well. Once again, its not enough to have shared community clouds with loads of virtual redundancy, but shared virtual servers, applications, and information repositories as well. Redundant IT data centers, along with their information, logistics, mission, and highly specialized intelligence gathering systems across hundreds of DoD agencies and departments is untenable, and commonized virtual data centers must be developed, which otherwise hinders mission effectiveness.

27. Da·ta • Cen·ter • Con·sol·i·da·tion (*dā'tə • sĕn'tər • kən-sŏl'ĭ-dā'shən*) Fuse, merge, blend, mingle, combine; To de-duplicate redundant data centers into shared cloud computing platforms when operating innovatively new products and services

- ✓ Deduplicate redundant IT data centers.
- ✓ Deduplicate redundant IT data center assets.
- ✓ Deduplicate redundant IT data center applications.

A similar example of DoD cloud computing principles is data center consolidation. The DoD is sprawling with small, medium, large, and monolithic traditional brick-n-mortar IT data centers. These may be a rack sitting in the corner of a conference room, a hallway closet, or a small server farm in the middle of a building. With hundreds of office buildings per agency, this results in thousands of individual brick-n-mortar IT data centers. Commercial cloud services offer a unique opportunity to virtualize individual racks, closet configurations, mini-data centers, large mission data centers, and monolithic high-performance data centers. Not only should DoD agencies, departments, business, and mission functions consolidate their IT data centers into virtualized commercial clouds, but the DoD agencies should look for opportunities to consolidate and share their virtualized IT data centers as well. The bottom line is rather than spawn hundreds of thousands of specialized virtual servers and data centers in commercial and on-premises clouds, which is immensely expensive, instead look for opportunities to reduce redundancy, optimize costs and resources, and mission effectiveness as well. The cost, risk, and timelines for operating thousands of redundant traditional IT brick-n-mortar data centers is enormous, costs can still get out of hand with individualized virtual data centers, and the annual $733 DoD budget is simply not enough to manage the exponential costs of technology obsolescence and entropy, much less per second, minute, or hour charges for virtual IT data centers in commercial or on-premises clouds. Many first-generation government cloud computing initiatives involve migrating hundreds and sometimes thousands of legacy information systems and repositories to virtual IT data centers and then data mining them using ad hoc streaming services. While this may seem like a painless process, the better choice is information system consolidation, reengineering, refactoring, or simply designing consolidated, replacement, or innovative minimum viable products (MVPs) instead. The per cost usage of migrating hundreds of legacy applications and exabytes of legacy data is simply economically prohibitive, not only for the largest DoD agencies, but certainly for the small and medium-sized ones as

well. It's simply grandiose and unwise to wire thousands of legacy systems together using Tasktop due to the per hour usage rates of most commercial cloud providers or do big bang consolidation initiatives. But, perhaps legacy IT data centers, information and mission systems, and their attendant repositories can be consolidated in phases over a period of years (i.e., 15,000 to 7,500, 7,500 to 3,750, 3,750 to 1,875, 1,875 to 938, 938 to 468, etc.). It's almost guaranteed that 95% of legacy systems and their data are not even needed, and some government agencies still have operational mission critical IT data centers, information systems, and repositories designed in the 1960s, 1970s, and 1980s, which is utterly ridiculous!

28. Da·ta • Stan·dard·i·za·tion (*dā'tə • stăn'dər-dī-zā'shən*) Consistent, normalized, customary, conformant, integrated; To de-duplicate redundant data and information types, models, and repositories across shared cloud computing platforms

- ✓ Deduplicate information data centers.
- ✓ Deduplicate information repositories.
- ✓ Deduplicate information reporting.

A common example of DoD cloud computing principles is data standardization. The exponential use and crisis of traditional DoD information systems has been known since at least the 1990s. The DoD hired armies of data analysts and scientists, engineers, and architects to define common data models beginning in the 1990s. The hope was that with standardized data models, schemas, and individual object specifications (i.e., tank, missile, aircraft, vehicle, etc.), perhaps information and data could be easily or at least more easily transported between IT data centers, applications, and repositories. However, just as the DoD has gotten out of the business of legislating IT delivery models, frameworks, and lifecycle standards, the DoD is also getting out of the business of legislating data models. With more and more IT outsourcing, commercial IT service providers, and commercial products and services, the DoD has traded off commonality, interoperability, and interchangeability for initial cost reduction, speed, and acquisition efficiency. Unfortunately, this has resulted in a veritable explosion of uncommon, non-interoperable, and non-interchangeable commercial information repositories. Oftentimes, the commercial enterprises themselves don't do a good job of information engineering and often embed information and data in the code itself (i.e., sort of like hardcoding credentials in a computer program instead of retrieving it from an identity management system for expediency's sake). The bottom line is that information systems, their repositories, and their data are less interoperable than they've ever been before, in spite of the fact that the DoD spent billions of dollars on enterprise architectures over the last 20 years. Once again, the commercial industry's response for the smorgasbord of incompatible information systems, repositories, and databases, is simply to migrate them to commercial cloud services and data mine them using high-speed streaming services (how convenient). It would be like jamming a million people into a refugee camp and then identifying any one of them with an RFID tag in a fraction of a second. Instead of jamming legacy DoD information systems into cloud-based refugee camps, let's design sparsely populated neighborhoods with plenty of open space. While data standardization plays a big part in interoperability, interconnectivity, information sharing, and data mining, replacing redundant information systems, having industry self-regulate common data models, and deduplicating and consolidating disparate information repositories into shared regional systems is better. In other words, its far superior to have 25 national healthcare information systems instead of 2,500 disparate commercial information systems. The same is true for DoD systems. Stop developing and hoarding tens of thousands of legacy information systems, migrating them to monolithic clouds at great risk and expense, and expecting to data mine their information using real-time streaming services and keep them around forever that's an unnecessary hat trick.

29. Shared • In·for·ma·tion (*shâr'd • ĭn'fər-mā'shən*) Data, facts, details, knowledge, databases; To designate common data and information repositories across shared cloud computing platforms when operating innovatively new products and services

- ✓ Share information data centers.
- ✓ Share information repositories.
- ✓ Share information reporting.

A better example of DoD cloud computing principles is shared information. That is, instead of interconnecting incompatible information systems with application programming interfaces (APIs) or investing in expensive enterprise architectures and common data models, simply share IT data centers, information systems, and repositories when possible. In one instance, there were over 2,500 legacy systems in use by dozens of law enforcement agencies to monitor incoming and outgoing immigrants. It was virtually impossible to see an individual immigrant with a criminal record in the myriad of information systems. To add insult to injury, the immigrants were changing their names, appearances, and crossing at different ports of entry to ensure they would not show up twice in any one of thousands of databases (i.e., quite clever). In one database, their criminal record might show up clean with one name, in another as a petty thief with another name, and perhaps something more serious with yet a third name. The trick was simply to change credentials once an identity had been compromised with a particular criminal act. One agency's response was not to create a common data model for 2,500 disparate systems or consolidate them to, let's say, 250 or 25 shared information systems, but simply to link them together using APIs and query out information using a global query or data broker. The notion was that one could query out a name, combination of names, criminal histories, photographs, physical characteristics, or even identifying marks to determine if an individual immigrant had a serious criminal record in any one of thousands of information systems in order to deny entry at a single point of entry. Nice try, but after $10 or $15 billion dollars, the border initiative was simply canceled without yielding any value. Clearly, the superior answer is common information systems, repositories, IT data centers, etc. With commercial cloud services, this is even easier. A common law enforcement cloud can be created, data migrated into a single data model or a flat file format like Hadoop Distributed File System (HDFS), indexed using Google's Big Table, and queried out using Hadoop Map Reduce. Many commercial cloud service providers offer low-cost scalable databases like Mongo, DynamoDB, and many others. Even traditional relational database technologies are catching up and providing the scalability once offered by bleeding edge cloud database technologies. The advantage of full service scalable relational technologies is they offer rich ecosystems of data

modeling, querying, and reporting features that have to be built one line of code at a time using bleeding edge cloud services. There's simply no reason why the government cannot build a single cloud based, scalable customs database that can be used by law enforcement, immigration, civilian, DoD, and commercial enterprises such as airports. Now is the time to replace, reengineer, refactor, and consolidate our DoD information systems using low-cost, scalable commercial cloud services.

30. Mis·sion • Work·flow • Sys·tems (*mĭsh'ən • wûrk'flō • sĭs'təmz'*) Unique, specialized, prosecutorial, domain-specific, mission-critical; To apply non-physical cloud computing platforms for delivering innovatively new warfighting products and services

- ✓ Mission-specific production services.
- ✓ Mission-specific production platforms.
- ✓ Mission-specific production applications.

Yet another example of DoD cloud computing principles is mission workflow systems. That is, the ability to compose large scale global information systems by threading together well-architected domain specific systems with common storefronts. This has been the dream of most DoD agencies since at least the early 2000s (i.e., the ability to thread legacy systems together by common application programming interfaces and graphical user interfaces). Some DoD agencies have tried to group small ecosystems of new and legacy information systems together under common GUIs and query systems. Other DoD agencies have tried to group all major legacy systems together under one GUI and query system. Of course, the challenge was often scalability, availability, reliability, elasticity, interoperability, and networkability. Common problems were that legacy information systems were not scalable to more than a few hundred warfighters or a couple of hundred terabytes. Furthermore, network links and bandwidth were limited, and legacy systems were not designed for interoperability or API technologies. Sometimes, new central information systems were created to collect data from multiple legacy systems, which was simply untenable. One such project involved dozens of legacy systems, poor information systems engineering, and about a billion dollars, which was simply thrown away after about a decade. The DoD is replete with billion-dollar boondoggles to do the same thing. Again, the problem was traditional brick-n-mortar technologies, rapid technological obsolescence, lack of modern IT skills and talent, lack of scalability, poor Intranet performance, and lack of commercial cloud service providers. Today, most of these problems are gone. Commercial cloud services are scalable to at least the petabyte level, which IT from the early 2000s was simply unable to achieve, and today is scalable to the exabyte level and beyond. Of course, another problem was getting the data out with traditional microprocessors, which was solved with massive parallel processing high performance clusters using Hadoop Map Reduce-like technologies. That is, with traditional microprocessors running in parallel under a single backplane, thousands of microprocessors could sift through dozens of exabytes in fractions of a second. Of course, all of this was not possible in the early 2000s, or before that time. With these technologies in hand, modern programming languages, API technologies, commercial scalable virtual clouds, and high-speed intranet and Internet links, all of these problems have magically disappeared. Individual agencies can create dozens of domain specific clouds for specialized data types and simply string them together with storefront clouds. It's that simple and today's possibilities are limitless. It doesn't cost billions of dollars, it doesn't take decades, and risk of failure is decimated. It lends itself to lean-agile thinking and frameworks, low-cost business experiments and minimum viable products (MVPs), small batches, and limited WIP.

31. Least • Priv·i·leged • Ac·cess (*lēst • prĭv'ə-lĭjd • ăk'sĕs*) Lowest, minimal, tiniest, smallest, slightest; To develop, deliver, and operate innovatively new products and services at the lowest level of security privileges necessary to maximize their use

- ✓ Least privileged production services.
- ✓ Least privileged production platforms.
- ✓ Least privileged production applications.

A great example of DoD cloud computing principles is least privileged access. That is, the notion that DoD systems can be accessed by people with lowest possible security clearance. Most DoD data is unclassified and should be kept at that level for easy global access by warfighters. This also alleviates the necessity for DoD personnel to be onsite and is conducive to a virtual distributed remote telecommuting workforce. This is the wave of the future and DoD thought leaders at the highest level realize this, although people at the bottom of the DoD food chain are entrenched in traditional dogma from the last century when it comes to privileged access of DoD information. The goal is information sharing, consolidation, and commonization. The purpose is to string together mission systems and have them instantly accessible by global warfighters at the point of the mission. With the proper credentials, there is no reason why warfighters cannot access the information they need using laptops, tablets, smartphones, smart helmets, smart goggles, and other mobile wearable computers. The same is true of DoD government workers and contractors (buyers and suppliers). There is no reason why they can't work from any point and still have access to the information, data, and systems they need without having to commute long distances, occupy expensive buildings, and substantially reduce their quality of life (i.e., work-life balance). With commercial cloud service providers, all of this is possible (i.e., not only the notion of multi-domain clouds, but multi-privilege clouds). Of course, the goal is to reduce the data and information to lowest possible privilege level, secure it tightly, and allow access to people with properly verified credentials). Some of the largest global consulting firms have managed to achieve exactly that. Their corporate information exists in commercial cloud services, it's accessed by a global remote workforce, they use ironclad commercially available and multi-factor mobile app authentication technologies, they use common low-cost commercial-off-the-shelf laptops, and their end-point devices and clouds are leak-proof. That is, there is no data spillage and it cannot be extracted. It can be accessed, read, created, and shared on the commercial cloud, but it cannot be copied off to external storage technologies, it cannot be printed, and it cannot be emailed or uploaded to out-of-network cloud services. The corporate data is simply captive on their commercial cloud services with simple, low-cost, common, commercial-off-the-shelf devices and security technologies. At stake is billions of dollars in global revenues, razor thin profits, global rankings, and innovative ideas and intellectual capital that is highly unique to their consulting agencies. So, if a Big Six global consulting agency can operate in commercial clouds

with 200,000 global employees in 170 different countries, then the DoD can do the same at the least privileged level of access, while assuring the highest level of information security. It's time for the DoD to step out of the Stone Age!

32. Da·ta • Cen·ter • Sun·set·ting (*dā'tə • sĕn'tər • sŭn'sĕt'ĭng*) Close, recall, retire, discard, withdraw; To decommission and dispose of unneeded brick-n-mortar data centers when deploying virtualized cloud computing resources to replace them

- ✓ Decommission brick-n-mortar production services.
- ✓ Decommission brick-n-mortar production platforms.
- ✓ Decommission brick-n-mortar production applications.

A final example of DoD cloud computing principles is data center sunsetting. In other words, it's time for the DoD to pursue a course of action that includes using commercial cloud services, creating domain specific clouds, threading them together into mission workflow storefront clouds, and simply decommissioning redundant traditional brick-n-mortar IT data centers of all shapes and sizes. New and innovative information technology is simply emerging at an exponential rate of speed, the DoD simply cannot keep up, and its IT delivery models are based on manufacturing principles from 1900. These include IMSs, enterprise architectures, volumes of big-batch business requirements written decades ago, and 5-10-and-15-year multi-billion capital intensive projects with a high rate of acquisition failure. Furthermore, traditional brick-n-mortar data centers are based on designs and technologies from the 1990s; they simply cannot keep up with the rate of technological obsolescence and entropy; individual physical IT assets are obsolete before a purchase order can be made, approved, and fulfilled; system outages are a daily occurrence; and their attack surface is so large and vulnerable, they are subject to frequent compromises. Instead of continuing to throw good money at obsolete capital-intensive traditional brick-n-mortar IT data centers, its time to invest our dollars in low-cost, low-risk, and ready-made virtual cloud service providers. Once again, virtual cloud services lend themselves to lean-agile thinking and frameworks, rapid business experiments, small batches, limited WIP, short lead and cycle times, fast feedback, and endless rinse-and-repeat cycles until the optimal mission value point can be achieve (all at a sustainable work pace, work life balance, quality of life, and superior level of morale). Of course, lean-agile frameworks are critical to limiting WIP, utilization, and backlog (queues), because cut-throat capitalistic firms will simply drive cloud engineers to full utilization to maximize revenues while burning out knowledge workers like flies. Once the goal of capitalizing upon lean-agile thinking, commercial virtual cloud services, domain specific clouds, and mission workflow storefronts is achieved, then the inevitable process of sunsetting or decommissioning legacy traditional brick-n-mortar IT data centers, assets, information systems, and data repositories can begin. Let's stop trying to rescue our legacy systems and start trying to reengineer, refactor, consolidate, and create innovatively new ecosystems of domain-specific cloud-based MVPs and storefronts. Legacy systems are sunk costs in accounting terms, and their exponential rate of decay translates into an exponential devaluation of the national budget, DoD budget, and DoD IT acquisition dollar. Furthermore, simply migrating hundreds or thousands of legacy systems to commercial clouds is NOT the answer, where per minute, hour, and day costs will simply melt taxpayer acquisition dollars away like an ice cube in hell. Like Michael Hammer said, "Obliterate it, don't automate it!"

DoD Cloud Computing Summary

So, what have we learned from this short treatise on how to successfully transition to DoD cloud computing data centers. Firstly, we've learned that lean-agile thinking and ready-made lean-agile frameworks are ideal for transitioning to cloud computing. That is, the DoD should not continue to make the same old mistakes it has for the last 70 years and attempt to build over scoped, gold-fleeced, and gold-plated cloud computing data centers using 5-10-or-15-year long integrated master schedules (IMSs), enterprise architectures, and business requirements. The economics of enormous requirements batches are simply untenable, over 95% of system requirements are unneeded or defective, and the probability of hitting cost, schedule, and technical performance goals, much less security targets, is simply impossible. Global organizations throw trillions of good dollars at bad and failed IT initiatives each and every year for applying traditional thinking principles and practices and shunning lean-agile thinking principles, frameworks, and their associated practices. Again, most market, customer, and end-user (warfighter) requirements exist as tacit, hidden, and inexpressible needs that must be teased out a little bit at a time with small business experiments, minimum viable products (MVPs), and many rapid lean-agile rinse-and-repeat cycles. Secondly, we've learned that lean-agile thinking is insufficient in of itself, and investments in mission and warfighting systems based on the secret sauce of virtualized cloud computing principles and practices trumps investments in traditional capital-intensive brick-n-mortar IT data centers that are subject to rapid technological obsolescence, entropy, and exposure to unmanageable security vulnerabilities. Therefore, it's good that the DoD applies lean-agile frameworks, but bad for operating and maintaining traditional brick-n-mortar IT data centers.

32 Principles and Practices to Successfully Transition to U.S. DoD Cloud Computing Data Centers

1. **Lean-Agile Thinking**—Apply small batches of WIP-limited experiments to gradually tease out tacit, inexpressible needs
2. **Lean-Agile Framework**—Apply a simple lean-agile reference model to develop innovatively new products and services
3. **Lean-Agile Contract**—Apply legally-binding terms and conditions for collaboratively developing products and services
4. **Virtual Experiments**—Apply small batches of WIP limited hypothesis tests to gradually tease out tacit, inexpressible needs
5. **One-Team Culture**—Create an open, collaborative, cooperative, communicative, and teamwork-oriented environment
6. **Security First Mindset**—Apply information security principles early and often when creating products and services
7. **Commercial Clouds**—Acquire use of open, publicly available cloud computing resources, applications, and technologies
8. **Community Clouds**—Form, apply, and utilize shared cloud computing resources, applications, and technologies
9. **Regional Clouds**—Form, apply, and utilize geographically distributed cloud computing resources, applications, and tech.
10. **Mission Clouds**—Form, apply, and utilize specialized cloud computing resources, applications, and technologies
11. **Virtual Data Centers**—Apply non-physical cloud computing resources, applications, and tech. for products and services

12. **Virtual Development Servers**—Apply non-physical cloud computing platforms for developing new products and services
13. **Virtual Production Servers**—Apply non-physical cloud computing platforms for hosting new products and services
14. **Mirrored Environments**—Apply duplicate non-physical cloud computing platforms for development and operations
15. **Blue-Green Environments**—Apply parallel non-physical cloud computing platforms for development and operations
16. **Elastic Production Servers**—Apply dynamically scalable non-physical cloud computing platforms for products and services
17. **Fault Tolerant Servers**—Apply redundant, failover non-physical cloud computing platforms for products and services
18. **Transient Environments**—Apply temporary, on-demand non-physical cloud computing platforms for products and services
19. **Automated Environments**—Apply programmable non-physical cloud computing platforms for products and services
20. **Least Privileged**—Apply and allocate the least amount of cloud computing security privileges to products and services
21. **Virtual DevOps**—Apply on-demand, non-physical application lifecycle tools for development and maintenance
22. **DevSecOps Practices**—Apply application security practices to develop, deliver, and operate new products and services
23. **Containerized Microservices**—Apply encapsulated application architectures to design and maintain products and services
24. **Automated Monitoring**—Apply computerized performance surveillance of cloud computing platforms
25. **Automated Response**—Apply computerized action plans to restore, balance, and protect cloud computing platforms
26. **Commonized Data Centers**—Apply and designate shared cloud computing platforms for operating products and services
27. **Data Center Consolidation**—De-duplicate redundant data centers into shared cloud computing platforms
28. **Data Standardization**—De-duplicate redundant data and information types, models, and repositories across shared clouds
29. **Shared Information**—Designate common data and information repositories across shared cloud computing platforms
30. **Mission Workflow Systems**—Apply non-physical cloud computing platforms to deliver complex warfighting ecosystems
31. **Least Privileged Access**—Develop, deliver, and operate new products and services at lowest level of security privileges
32. **Data Center Sunsetting**—Decommission and dispose of unneeded traditional brick-n-mortar IT data centers from the 1990s

Thirdly, we've learned that there are hundreds of DoD agencies and departments and each one has thousands of legacy systems in their IT portfolios. This is due to a few closely related factors, such as having the largest annual operating budget in the world (i.e., $6 trillion U.S. and $733 billion DoD budget) and the fact that the Western hemisphere is rife with fiercely individualistic selfish-suboptimization. That is, cooperation, collaboration, and communication between individuals, teams, projects, programs, portfolios, and agencies is limited, if not prohibited by our culture, giving rise to millions of non-cooperating IT systems at a cost of trillions of taxpayer dollars. As a result of simply having too much money, or the willingness to spend too much money, and selfish suboptimization deeply embedded or tattooed in the psychology of Western engineers, the DoD insists upon building thousands of redundant and incompatible IT systems each year. Furthermore, it spends billions of dollars trying to save legacy IT systems and their information repositories created from 1950 to the present time, integrate them into new information systems, and transition their combobulations of old and new IT systems to virtualized cloud computing data centers. Nearly 80% to 90% of the annual DoD budget is spent keeping and transitioning up to 70-year-old IT systems to the cloud, creating new concoctions with alchemy by migrating them to clouds, and data mining their information repositories with real-time streaming services and continuing to operate them as-is in the cloud. If the DoD is willing to spend $600 billion saving useless information systems because they simply like to hoard legacy IT systems, then a veritable cottage industry of contractors is willing to invent new snake oil salves to help them do so in order to get a piece of the pie, form one person firms, and continue the curse of fierce Western individualism.

Fourthly, we've learned that it's in sufficient for DoD agencies to migrate their individual IT portfolios to commercial or on-premises clouds, but should instead seek synergies, economies of scope, and good old economies of scale when combining IT systems and clouds. That is, not just build and share community clouds, legacy IT systems, and their repositories, but build and share domain specific community clouds. There are just far too many disparate commercial, non-DoD, and DoD IT systems, and the possibility of integrating these systems and their data into a single whole is simply a pipe dream. Common data models must be formed, the infinite variety of redundant systems and repositories must be consolidated, and regional subportfolios of similar systems must be created. For instance, it may not be possible to build one single DoD healthcare information system, although we've unsuccessfully tried to do so for more than a decade using lean-agile frameworks, but it may be possible to consolidate and downsize the total number of DoD healthcare systems by an order of magnitude or two (i.e., 20 or 200, instead of 2,000+). More importantly, instead of just migrating legacy systems to the cloud, let's use the synergy of lean-agile thinking and state-of-the-art cloud computing technologies to refactor, reengineer, and create new minimum viable products (MVPs) in weeks and months instead of a cacophony of maximum viable products at a cost of billions of dollars and decades. Unfortunately, there is still a lot of non-invented-here (NIH) in the DoD and IT systems engineers are simply allergic to the notion of cloud computing and break out into hives when they hear it, due to lack of understanding, and psychological attachment to traditional brick-n-mortar IT systems, rapid technological obsolescence, unmanageable security vulnerabilities, frequent system outages, and failing DoD missions.

Okay, so what's the bottom line? Firstly, the DoD needs to aggressively adopt lean-agile thinking values, principles, frameworks, practices, metrics, tools, AND technologies at all levels. Commercial cloud services are the ultimate form of lean-agile technology, and when combined, serve as an unstoppable combination for cranking out new minimum viable products (MVPs) at lightning speed to seize the upper hand when it comes to the infinite variety of ever evolving global military threats. This is true for conventional battles, air campaigns, space systems, missile defense, remote vehicles and drones, logistics and resupply systems, command and control, acquisition and contracting, administration and business, intelligence gathering, and, yes, even healthcare administration. When it comes down to it, lean-agile thinking is all about temporary, throwaway business experiments, and the DoD needs to make the shift from hoarding hundreds of thousands of 70 year old legacy systems that cost trillions of dollars to maintain and wake up to the fact that small teams of cooperating, communicating, and collaborating engineers can quickly compose new mission systems in hours, days, and weeks, discard them, and begin again without the cost of saving legacy systems. It's a mindset, but old habits die hard, and DoD engineers continue clinging to integrated master schedules (IMSs), enterprise architectures, and business requirements, while old IT becomes obsolete and insecure at an exponential rate of speed.

Further Reading

- Andrews, D., et al. (2019). *DoD cloud adoption*: *The department's cloud strategy sets a modern vision for cloud adoption, but jedi rfp will not achieve that vision*. Alexandria, VA: ITAAC.
- Artasanchez, A. (2021). *Aws for solutions architects: Design your cloud infrastructure by implementing devops, containers, and amazon web services*. Birmingham, UK: Packt Publishing.
- Banfield, R., Lombardo, C. T., & Wax, T. (2016). *Design sprint*: *A practical guidebook for building great digital products*. Sebastopol, CA: O'Reilly Media, Inc.
- Blum, M. (2014). *TechFAR*: *Handbook for procuring digital services using agile processes*. Washington, DC: U.S. Digital Services.
- Dantas, V. (2021). *Architecting google cloud solutions*: *Learn to design robust and future-proof solutions with google cloud technologies*. Birmingham, UK: Packt Publishing.
- Deasy, D. (2020). *Interim guidance for the implementation of the department of defense cloud strategy*. Washington, DC: Dept. of Defense.
- Denning, S. (2018). *The age of agile*: *How smart companies are transforming the way work gets done*. New York, NY: Amacom.
- DISA. (2017). *Department of defense cloud computing security requirements guide*. Washington, DC: Department of Defense.
- Duvall, P. M. (2021). *Enterprise devops on amazon web services: Releasing software to production at any time with aws*. Boston, MA: Addison-Wesley.
- Geewax, J. J. (2018). *Google cloud platform in action*. Shelter Island, NY: Manning Publications.
- International Standards Organization. (2017). *Collaborative business relationship management systems*: *Requirements and framework* (*ISO 44001*). Geneva, Switzerland: International Organization for Standardization (ISO).
- Knapp, J. (2016). *Sprint*: *Solve big problems and test new ideas in just five days*. New York, NY: Simon & Schuster.
- Luca, M., & Bazerman, M. H. (2020). *The power of experiments*: *Decision making in a data-driven world*. Cambridge, MA: MIT Press.
- Lynn, T., Mooney, J. G., Rosati, P., & Fox, G. (2020). *Measuring the business value of cloud computing*. Cham, CH: Palgrave-MacMillan.
- Madamanchi, S. (2021). *Google cloud for devops engineers*: *A practical guide to site reliability engineering*. Birmingham, UK: Packt.
- McHaney, R. (2021). *Cloud technologies: An overview of cloud computing technologies for managers*. Hoboken, NJ: John Wiley & Sons.
- Norquist, D. L. (2019). *DoD digital modernization strategy*. Washington, DC: Department of Defense.
- Odell, L. A., Wagner, R. R., & Weir, T. J. (2015). *Department of defense use of commercial cloud computing capabilities and services*. Alexandria, VA: Institute for Defense Analysis.
- Peters, H. J. (2019). *The department of defense's jedi cloud program*. Washington, DC: Department of Defense.
- Rico, D. F. (2016). *The 10 attributes of successful teams, teamwork, and projects*. Retrieved September 26, 2016 from http://davidfrico.com/teamwork-attributes-2.pdf
- Rico, D. F. (2016). *The 12 attributes of successful collaboration between highly creative people*. Retrieved February 29, 2016, from http://davidfrico.com/collaboration-attributes.pdf
- Rico, D. F. (2017). *U.S. DoD vs. Amazon*: *18 architectural principles to build fighter jets like Amazon web services using DevOps*. Retrieved January 26, 2017, from http://davidfrico.com/dod-agile-principles.pdf
- Rico, D. F. (2018). *Lean & agile contracts*: *21 principles of collaborative contracts and relationships*. Retrieved June 29, 2018, from http://davidfrico.com/collaborative-contract-principles.pdf
- Rico, D. F. (2018). *Using SAFe 4.5 to transform a $200 million U.S. healthcare portfolio*. Retrieved November 19, 2018 from http://davidfrico.com/safe-case-study-ii.pdf
- Rico, D. F. (2019). *32 attributes of successful continuous integration, continuous delivery, and DevOps*. Retrieved September 27, 2019, from http://davidfrico.com/devops-principles.pdf
- Rico, D. F. (2019). *Business value of lean leadership*: *22 Attributes of successful business executives, directors, and managers*. Retrieved April 27, 2019, from http://davidfrico.com/rico19a.pdf or http://davidfrico.com/lean-leadership-principles.pdf
- Rico, D. F. (2019). *Evolutionary architecture*: *24 principles of emergent, organic, and highly adaptive design*. Retrieved September 3, 2019, from http://davidfrico.com/evolutionary-architecture-principles.pdf
- Rico, D. F. (2019). *Piloting the scaled agile framework (SAFe) in a top 10 U.S. energy firm*. Retrieved May 15, 2019, from http://davidfrico.com/x-safe-case-study-iii.pdf
- Rico, D. F. (2020). *32 principles and practices of highly successful SAFe implementations*. Retrieved February 16, 2020, from http://davidfrico.com/safe-principles.pdf
- Rico, D. F. (2020). *Business value of lean thinking*: *Capitalizing upon lean thinking principles to rapidly create innovative products and services*. Retrieved January 29, 2020, from http://davidfrico.com/rico20b.pdf or http://youtu.be/wkMfaPAxO6E
- Rico, D. F. (2020). *Using large solution SAFe in a high-profile U.S. civilian public sector agency*. Retrieved January 29, 2020, from http://davidfrico.com/y-safe-case-study-iv.pdf
- Rico, D. F. (2021). *32 principles and practices for a highly successful agile contract statement of work (SOW)*. Retrieved March 22, 2021, from http://davidfrico.com/agile-sow-principles.pdf
- Rico, D. F. (2021). *32 principles and practices for maximizing the ROI of SAFe program increment (PI) planning*. Retrieved March 14, 2021, from http://davidfrico.com/safe-roi-principles.pdf
- Rico, D. F. (2021). *32 principles and practices for successful distributed lean-agile programs, projects, and teams*. Retrieved May 7, 2021, from http://davidfrico.com/agile-distributed-principles.pdf
- Rico, D. F. (2021). *Business value of organizational agility*. Retrieved March 26, 2021 from http://davidfrico.com/rico21a.pdf or http://davidfrico.com/value-of-business-agility.pdf or http://youtu.be/HOzDM5krtes
- Rico, D. F., Sayani, H. H., & Sone, S. (2009). *The business value of agile software methods*: *Maximizing ROI with right-sized processes and documentation*. Ft. Lauderdale, FL: J. Ross Publishing.
- Ries, E. (2017). *The startup way*: *How modern companies use entrepreneurial management to transform culture and drive long-term growth*. New York, NY: Crown Publishing.
- Schrage, M. (2014). *The innovator's hypothesis*: *How cheap experiments are worth more than good ideas*. Boston, MA: MIT Press.
- Shanahan, P. M. (2018). *DoD cloud strategy*. Washington, DC: Department of Defense.
- Soh, J., Copeland, M., Puca, A., & Harris, M. (2020). *Microsoft azure: Planning, deploying, and managing the cloud*. New York, NY: Apress.
- Thomke, S. H. (2020). *Experimentation works*: *The surprising power of business experiments*. Boston, MA: Harvard Business Review Press.
- Vacca, J. R. (2021). *Cloud computing security: Foundations and challenges*. Boca Raton, FL: CRC Press.
- Wittig, A., & Wittig, M. (2018). *Amazon web services in action*. Shelter Island, NY: Manning Publications.

CASE STUDIES OF SUCCESSFULLY TRANSITIONING TO U.S. DOD CLOUD COMPUTING DATA CENTERS

- On Premises DoD Analytic Cloud. *This was one of the earliest technology demonstrations of transitioning to cloud computing when few people even knew what that meant. This was a large DoD agency where the lion's share of its annual operating budget was dedicated to operating and maintaining thousands of legacy IT systems and reinventing its IT infrastructure at the same time. Prior to about 2000, it had been frozen in time with little to no IT in use, save that of a few large mission systems. As a result of the Internet explosion of the 1990s, it finally caught up and devoted its entire IT budget to modernizing its infrastructure, business, and mission applications with Web technologies. This is not to say it didn't have some advanced Internet research prototypes in use for mission purposes in the 1990s, but these were few and far between although very effective. Many of its enterprise and mission systems, which were often convoluted, relied on relational database technologies from the 1980s and 1990s. However, this agency had a hard time scaling any one system beyond about 100 terabytes, even with custom rack elevation server towers populated with sophisticated high-performance components. They even tried expensive hardcoded data warehouse engines to no avail. They simply couldn't scale effectively beyond 100 or 200 terabytes and these systems certainly couldn't interface to the rest of the enterprise. After spending a few billion dollars throwing good money at bad database projects, it discovered cloud technologies around 2010. That is, frustrated with failed RDMS projects, it visited Silicon Valley Internet giants and discovered what the Dot Coms were playing with. Google pioneered many software solutions around 2000, namely HDFS, BigTable, and MapReduce. However, they still needed high performance clusters to host these open source software components. So, the DoD agency procured a custom high-performance cluster, loaded Google's cloud technologies, and "wa la" they could now store and retrieve Petabytes of mission data in fractions of a second for about $10 million dollars per repository. This was more than an order of magnitude less than their latest failed RDMS project. These technologies were bleeding edge, they were stripped down MVPs, and they didn't play well with other enterprise applications. Therefore, the data was stranded in the on premises point solution, but it was a very valuable proof-of-concept all by itself after decades of settling for simple mission systems that were counterproductive for military purposes and failing to apply RDMS technologies to the scale, scope, and size of their military missions. Lean-agile frameworks were viciously applied, and its scale, scope, and size was kept to the most essential mission requirements (i.e., store and retrieve petabytes of data in a few seconds or less). Although the high-performance cluster was very resilient by default, there was no fault tolerant cluster, and the hardware technology was soon obsolete in a matter of 90 days. Google's software technologies were also obsolete, which is why they were giving them away as open source software components at no cost. However, it established the roadmap for future cloud initiatives.*
- On Premises DoD Storage Cloud. *This agency had a lot of mission data it collected over the decades. Much of it was held captive in siloed enterprise information systems as well as field sites sprawled around the globe. In order to help ensure continuity of operations, it decided to build a single large distributed storage cloud for collecting all of its past, present, and future mission data in a single family of nationally distributed IT data centers. However, the distributed data centers were to act as a single information system that could be treated as a single large repository and queried from individual warfighter personal computers. It was a nice thought. It also visited Dot Coms in Silicon Valley to learn how they developed large storage clouds for storing and distributing large volumes of data for millions of global customers in real-time. They discovered peer-to-peer technologies in use by the music industry for storing and distributing audio files and decided to try that out. Most rack elevations could store a few terabytes and the goal was to have hundreds of these rack elevations at each one of multiple sites and have redundancy of data across the national fabric of data centers using peer-to-peer technologies, just like the folks in Silicon Valley. That is, if any one of the sites or servers was compromised, there was at least one redundant copy of the data element needed at any one time to warfighters. Although it was a cute multibillion dollar toy, it was subject to many limitations (i.e., primarily national distribution). That is, if a site went down, the peer-to-peer open source software would simply start shifting petabytes across DoD network links, saturate them, and freeze the individual data centers and national network hubs. In other words, it was neither scalable, reliable, available, nor redundant. Engineers had to monitor national network links and servers 24 hours a day and shut down network devices to prevent the saturation of DoD networks if a server failed and the peer-to-peer software tried to replicate its data. At least they had the storage cloud populated, but they hadn't solved the problem of automated continuity of operations. Then cloud technologies arrived on the scene about 2010. They considered primitive technologies like HDFS, BigTable, and MapReduce, but realized they were too primitive for interfacing, interlacing, and integrating them into a larger enterprise mission system fabric (i.e., it was cost prohibitive to interact with them through a casual GUI). Then, better cloud technologies emerged like Cassandra that offered the scalability of HDFS with the application interfaces of RDBMS technology. Using the storage servers, they simply installed Cassandra one data center at a time and moved the storage data to the new cloud fabric consisting of thousands of nation-wide storage servers. Cassandra also offered some redundant fault tolerant properties that primitive peer-to-peer technologies from the 1990s simply couldn't do. Furthermore, it could be indexed with BigTable, interfaced to enterprise applications, and serve as a better, scalable, high-performance enterprise storage cloud than the initial solution. A few billion dollars later, they began building scalable community clouds.*
- Community DoD Cloud Initiative. *By the mid-2010s, this DoD agency was now getting comfortable with scalable on-premises hardware and software cloud technologies for general purpose mission use. Therefore, it identified dozens of mission critical applications and asked them to migrate off of their traditional IT data center hosts (i.e., often times simple rack elevations in closets and conference rooms) and onto larger shared on-premises clouds. General purpose cloud service providers like Amazon Web Services were just entering their golden age and Microsoft and Google were just entering the marketplace of general-purpose cloud service providers after years of building custom clouds for their own market, customer, and end-user applications (i.e., Googlesearch, Gmail, Gmaps, etc.). AWS began generalizing their entire platform beyond the ability to engineer scalable storage clouds to a globally scalable and elastic general purpose application hosting system of systems. Furthermore, AWS was now selling its hardware and software components to commercial and government clients to build their own on premises general purpose clouds. Heck, AWS would develop and maintain non-DoD and DoD on premises clouds running their hardware and software fabric for you if you paid them enough money (and many government agencies with highly sensitive enterprise data took them up on that offer, including the DoD). However, this DoD agency was still using primitive storage clouds based on HDFS and Cassandra and was nowhere near the maturity of AWS for hosting general purpose enterprise applications. Microsoft's Azure was just getting started and it seemed like they would favor Azure over AWS, while the rest of the DoD began striking deals with AWS for custom on-premises clouds (along with non-DoD agencies). This particular DoD agency was still in the traditional mindset of building custom enterprise hardware and software systems, a cultural element dating back to the 1950s, and wanted to build custom storage clouds for the rest of the DoD in direct competition with AWS, Azure, Google, Oracle, Rackspace, and many others. It simply had no sense that storing and retrieving exabytes of mission data in fractions of a second or creating large storage clouds was only a fraction of the needs for DoD cloud computing. In other words, the DoD also needed to host common commercial applications such as Microsoft Office, billing and payroll systems, healthcare data, and other common enterprise information systems. It wasn't just a matter of jamming a few exabytes of data into a non-interoperable storage server and running BigTable and MapReduce to find a needed data element in a few seconds. However, their advantage is that their custom on-premises storage clouds were expressly designed for DoD mission data, whereas commercial clouds could do it too, but the notion of mixing administrative and mission data on general purpose clouds was a bit suspect (and still is to this day). Although some large DoD agencies have successfully stood up complex mission clouds using AWS, and are now expanding into other cloud services, mission data information security is still a politically supercharged issue.*