

BlueSpace Blog – December 2011

Cloud. You have been Assimilated. – Part Two.

What Does Cloud Mean, Exactly?

By Justin Marston

In case you missed out on the last blog, I should mention first off that this is the second of a 3-blog series all relating to that ambiguous word we hear thrown around so often: 'Cloud.'

What was 'Cloud' again?

And back to the buzzword. Whereas parallel computing, clustering, grid architectures and virtualization have reasonably well-specified and agreed boundaries and definitions, 'cloud' is a little bit, well...fluffier.



There are many different themes in cloud computing – some appropriate, and some less so. The more relevant ones I would pick would be:

- **Networked:** A cloud must encompass many nodes, such as client devices accessing a web application that is hosted remotely, or computers in a peer-to-peer cloud.
- **High Availability:** Generally, cloud services have a degree of redundancy and no single point of failure (achieved via clustering and/or grid techniques).
- **Utility Computing:** The concept of being able to pick up and use computing resources like turning on the light switch; this tends to mean burstable, on demand with 'multi-tenancy.'

By 'multi-tenancy' here, I mean that multiple organizations can use the same set of compute resources, each of them bursting up and down as necessary. These organizations might be different departments in a company with a private cloud, or different companies in a system like AWS.

So these are the ones I've picked. Of course, given that it's the latest buzzword (and this comes back to my photo of an advert in an airport mentioned in Part One of this blog series), the world and his dog (and all it's fleas) all have cloud strategies. And there are [cloudlets](#) (Microsoft), [cloud 2.0](#) (Oracle), [social cloud](#) (IBM), [enterprise service cloud](#), [cloud service bus](#),...



In addition to themes and leveraged technologies/architectures, there are also different levels of cloud service – often described as ‘layers.’ Different people vary in the exact number and names of the layers in the ‘cloud stack,’ but the variances are small. In the image shown:

- **Client:** This is essentially a thin client, or some type of client device that is useless without the services from the ‘cloud.’ Some people really see thin clients as an access way to the cloud, but not a layer in its own right.

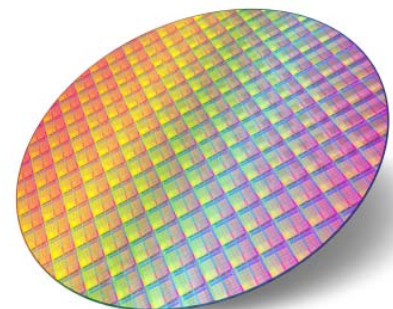
- **Application:** Also sometimes described as Software as a Service, this can include web applications – apps that do not need to be installed on the local computer.

- **Platform:** This layer typically abstracts the infrastructure services away from the developer. For example, Google’s [App Engine](#) allows developers to focus on their business logic, calling an API for storage, databases, etc. A developer can deploy an application to a cloud platform in a similar way to how they might have deployed a Java application previously, not worrying about how their application correlates to the underlying infrastructure. Hadoop and HBase can be part of a cloud platform.
- **Infrastructure:** This often includes a virtualization layer to abstract away from the specific underlying hardware, though if you are writing an application in a MapReduce style on something like a large Hadoop cluster, you don’t necessarily need a virtualization layer. I would bet serious money that Google runs GFS on bare metal.
- **Server:** The lowest layer, this is the physical hardware, and is typically commoditized cheap servers. Sometimes the server hardware can be geared specifically towards cloud environments (I would cite Cisco as an example here).

Most apps I download to my iPhone today have some ‘cloud’ element to them – I can’t think of many that have zero reach-back. Facebook and web email clients are perhaps the most commonly used pure cloud applications. The speed and availability of internet-peered networks has made remote backup to the cloud (e.g. [Carbonite](#)), and group file storage in the cloud (e.g. [Dropbox](#)) a reality. It is truly astonishing when I sit back and think about it – while I’ve been writing this blog article, iTunes on my MacBook Pro has downloaded my purchase of the *Band of Brothers* series – that’s like 30GB of data.

Anyone remember 14.4Kbit baud? Now I’m on something like a 10Mbit Time Warner connection – at my house. AT&T U-verse goes up to 24Mbit these days.

One of the more interesting projects I read about recently was an Intel Corp research project aimed at turning the cloud concept on its head, or on a chip, in fact. The [Single-chip Cloud Computer](#) currently has 48 cores, and will scale to 100



cores and beyond. But to quote Intel, it's kind of like ESX on a chip:

In a sense, the SCC is a microcosm of a cloud datacenter. Each core can run a separate OS and software stack and act like an individual compute node that communicates with other compute nodes over a packet-based network.

People sometimes talk about chasing ambulances to go after the current pain points. A more appropriate metaphor may be that vendors and integrators chase armored trucks carrying money. It's easy to figure out which buzzwords are carrying dollars, and two right now (especially in the government, but also in the corporate world) are 'Cyber' and 'Cloud' – 'Cost savings' is another one that recently left the compound in the government arena, and is picking up speed.

So we used to have 'IT Security' and 'Information Assurance', but today it's 'Cyber Security.' And that thing that used to be your data center? Now it's your 'Cloud.' Or your 'Private Cloud.'

Government Clouds

Within the defense and intelligence community, there is a set of unique requirements for clouds – especially given that a significant proportion of the data is classified, and so it has all kinds of handling restrictions. You aren't going to see that data on EC2/S3 anytime soon.

I am seeing three categories of vendors in this space:

- **Big Infrastructure:** The typical IT infrastructure providers are now all in the "cloud" business – including IBM, HP, Oracle, Cisco, EMC, Dell, etc.
- **Cloud Specialists:** The likes of Amazon and Google are increasingly hanging around the [DISA](#) folks at conferences, and I wouldn't be surprised if we see business going on there, moving unclassified data to 'public clouds' or hosting private versions of their platforms on government networks. I would also include administration tools vendors, such as [Cloudera](#) – I have seen them speaking at quite a few conferences recently (and interesting, [In-Q-Tel](#) is one of their investors).
- **Cloud Service Providers:** A few of these are starting to emerge, offering hosted data centers and ultimately cloud services, designed for classified environments. [Terremark](#) (a subsidiary of Verizon) is an obvious example, though AT&T and Harris are also active in this space. It's likely more system integrators will get into this business as well (I'd bet they already have).



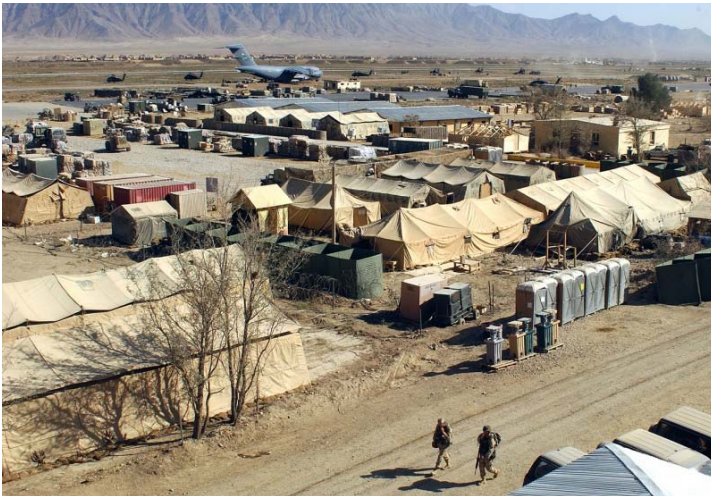
Interestingly, Terremark may end up competing with DISA. If budgets do get tight and departments are empowered to make their own acquisitions, it seems no different than

choosing which system integrator you want to use, but it may become more modular and service driven (as opposed to the unit of billing being ‘bodies’).



As for DISA, it is aggressively trying to move itself to providing ‘enterprise services’ for the community from its [Defense Enterprise Computing Centers](#) (DECC) – provided from the ‘cloud.’ DISA has been moving all the services (starting with the US Army) to its Enterprise Email service (essentially a hosted Microsoft Exchange environment). In our space, it has also issued several RFIs asking for proposals for cross domain guards and specific cross domain applications (email, chat, etc.) – looking to move from decentralized point services to centrally managed guard systems that organizations across the community can leverage.

A new twist we have started to hear recently is a distinction between ‘enterprise clouds’ and ‘tactical clouds.’ This really comes down to size and bandwidth. For US based users and services, you can really deliver on a vision of a single ‘enterprise cloud’ (apart from the political and funding boundaries). The network is fast enough to do it – you are running across dedicated fiber with huge backbone data rates.



That’s not true in theater (expeditionary/tactical forces). You might have decent local connectivity between nodes, but a slow backhaul via satellite to the ‘enterprise.’ In this expeditionary context, the Army defines tiers for levels of infrastructure – tier 1 being dismantled or a squad level, tier 2 being vehicles and convoys, and tier 3 being forward operating bases (from small company level clusters to huge bases with data centers, like Bagram).

I think cloud concepts will eventually touch some of these sites too, especially the tier 3 data centers. Imagine if you had ‘blocks’ that you could easily add to and split without worrying about the applications on them. Imagine if you could organically grow your ‘tactical cloud’ just by buying server infrastructure, and plugging them in.

I think Cisco’s thoughts on containerized data centers point in this direction, but I’m not aware of the Army and DoD trying to impact their application layer in a tactical context so that the applications can be ‘cloud optimized.’ For example, Microsoft Exchange is not set up this way right now – it is specific to a set of servers, and you can’t split off a block and have all the necessary services set up on it, scaled to the unit that’s going off to do the mission.

If I could just unplug a blade enclosure, ram it into my Humvee, and rapidly provision a set of apps running on it that I need for my mission – running ‘striped’ across all the blades – that could be pretty cool. ‘Elasticity at the edge.’ Tactical computing infrastructure then becomes

more like ammo – you grab a box, and then you head out. When you get back, you reinsert back into the ‘mother cloud,’ and it syncs. Perhaps I should write a separate blog article on that sometime, but for now, I’m going to get back on topic.

Multi-Level Clouds

I have written about many aspects of multi-level clouds before under different blog topics, but this time, I’m going to structure my categorizations and descriptions (of the technologies that are in the public domain today) around the five layers of the cloud described above:

- **Client:** There are several trusted workstation options out there, and who knows, maybe eventually there will be tablets and phones too. For cloud based clients there are Oracle (Sun Ray / Solaris) and TCS (third party / SELinux) options. They access multiple Microsoft Windows desktops from each user, hosted on separate Windows Terminal Servers running on the different classified networks.
- **Application:** BlueSpace exists in this layer, providing MLS mashup applications that are ‘orchestrated MILS’ web applications with user interfaces that span the different network domains. Of course, there are all the normal web applications that can run at a single level across the MLS infrastructure too. Cross domain data guards can act as an enabling technology for connecting cloud applications running at different security levels (and there are a plethora of guards out there). From a provisioning perspective, there are already a bunch of AppStore projects going on in the DoD and IC at a single level, and I think you will see BlueSpace plugging into those to provide an MLS appstore – provisioning of single level apps and multi-level apps across multiple logical network domains running at different security levels.

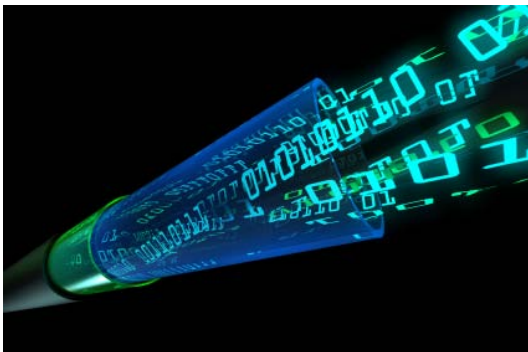


- **Platform:** Broadly, I think you will see regular commercial platform technologies used at a single level, relying on layers further down the stack to do the Mandatory Access Control (MAC) separation, just like I doubt I’ll ever see an MLS Microsoft Windows in my lifetime. Having said this, the most interesting project in this space, has been NSA’s recent releases to Hadoop and related infrastructure. According to some folks I know, NSA has been coveting true grid based architectures for some time, and considering how to increase robustness for fine grained access controls (analogous to FLASK in SELinux). NSA recently [contributed code back](#) to the Apache Foundation in [Accumulo](#) – a BigTable/HBase clone. The ambition isn’t to take over everyone’s brains with clever trapdoors (I just read some user comments on the announcement) – it’s an appetite to see commercial support for a platform NSA would like the DoD/IC to be

able to use internally. This reflects NSA's broader 'commercial for classified' strategy. A related effort is going on right down the street from us at the University of Texas, to introduce labeling into [MapReduce with SELinux](#) (surprisingly, [Google helped fund this](#)). I'd probably include Oracle's [Label Security](#) database in this platform layer, and also research efforts to provide a labeled [Java VM](#), though I don't think the latter ever reached a production system.

- **Infrastructure:** Today, there isn't an MLS capable technology solution for virtualization in the infrastructure layer, but that's changing. The AF is pursuing a CMATH ATD that should integrate the separation techniques used for XenClient back into XenServer, and provide a PL3 or possibly PL4 capable virtualization layer. Given this, and my conversations with people from a certain government agency, I think it's a matter of time before VMware follows suit with ESX – after all, robust multi-tenancy must be a requirement for the infrastructure layer in commercial clouds. In addition, some of the RTOS vendors have been looking at this space, as they provide separation kernels that run underneath hypervisors. You could also argue that zones in Solaris 10 TX (and a similar container-based mechanism that Red Hat has plugged into SELinux) have similar attributes, but I think they belong in the Server layer. From a system administration perspective, BlueSpace has already evangelized the idea of an [MLS NetOps](#) dashboard, and this is really a subset of a broader MLS system provisioning requirement. I think you'll see us active in this space, creating mashups (that span network enclaves) of existing provisioning web apps.
- **Server:** So as mentioned above, Solaris 10 TX and Red Hat SELinux do have features aimed at very large system clusters, which include labeling of UNIX containers. There are also networking technologies that might be considered for inclusion in the server layer, described below.

There are other technologies at early stages of development in several of these categories, but they aren't public yet. I think you'll continue to see more technologies developed in the MLS arena, around the drivers of cost savings, integration with virtualization, the continually increasing number of coalition networks, and the need for better network agility. I wrote about this recently in a blog article titled [Networks on Demand](#).



There's a final area in the MLS space that isn't really included in any of the five layers of the cloud as described above. In most cloud environments today, there is a single physical network. Corporations already use VLANs and VPNs to segregate cloud enclaves in the context of multi-tenancy and hybrid/private clouds, and can then have [federated](#)

[cloud](#) environments with limited connectivity between cloud domains. Although none of the cloud models call this out explicitly, I would put the networking piece in the bottom layer – the 'server' or 'hardware' layer.

In an MLS environment, you really need the ability to run multiple network channels down a single Ethernet cable so you don't end up with an inconceivable number of network cables everywhere. Having to rewire things physically every time you want a new 'private cloud domain' would be a real pain in the ass. Fortunately, there are some technologies out there to help with that, especially 'black channel' and 'brown channel' technologies. In this space, OIS has been working a project linked to CENTCOM, called [OBI](#). PACOM has a JCTD right now called CANDID, which is also active in this space. To quote from the [OSD summary](#):

CANDID will demonstrate the integration of Virtual Secure Enclaves (VSEs) inside existing tactical networks to enable network defense-in-depth and ensure Command and Control (C2) capabilities...

As I think about how clouds will be deployed in an MLS environment, I think you will see different levels of separation for different Protection Levels. There are many more COI than releasability caveats, and many more releasability caveats than security levels. So here are some closing thoughts:

- I think that for larger systems, you will likely see hardware-based separation on the backend for the foreseeable future for different classification levels. After all – there's really only 3 major ones – Unclassified, Secret and Top Secret. In a tactical environment, however, where size and weight is at a premium and user sets can be small clusters with poor long haul comms, I think you may see multi-tenancy of classification levels on single hardware, e.g. through trusted hypervisors.
- For COIs and caveats, especially given projects like CANDID at PACOM, I think you will see increasing use of virtualized channels and shared hardware for PL3 type separation within a classification level, using separation at the infrastructure layer. We might even see it at the platform layer eventually, especially for COI.