Annals of Emerging Technologies in Computing (AETiC)
Vol. 4, No. 5, 2020

Review Article

Hybrid Cloud SLAs for Industry 4.0: Bridging the Gap

Lubna Luxmi Dhirani* and Thomas Newe
Department of E&CE, University of Limerick, Limerick, Ireland
lubna.luxmi@ul.ie; thomas.newe@ul.ie
*Correspondence: lubna.luxmi@ul.ie

Received: 6th November 2020; Accepted: 11th December 2020; Published: 20th December 2020

Abstract: Hybrid Cloud Service Level Agreements (SLA) comprises of the legal terms and conditions for the cloud contract. Even though all the service level objectives, metrics and service descriptions are clearly outlined in the cloud SLA contract, sometimes vendors fail to meet the promised services and confusing terms lead to tenant-vendor cloud legal battles. Hybrid Cloud involves two different cloud models (public and private) working together, applications running under the hybrid cloud are subject to different availability sets, functionality and parameters developing SLA complexity and ambiguity. The new manufacturing environment (Industry 4.0 concept) is based on a fully connected, intelligent and automated factory, which will highly be dependent on cloud computing and IoT-based solutions for data analytics, storage and computational needs. In situations where Hybrid cloud services are not defined and managed properly may result in Industrial-IoT data security issues leading to financial and data losses. This paper discusses various aspects of the cloud service level agreement in Industry 4.0 for better understanding and implementation and puts a light on the issues that arise out of imprecise statements.

Keywords: ***Industry 4.0; Cloud Computing; Service Level Agreements; Data Security; Quality of Service***

1. Introduction

Cloud Computing is well-known for efficiently providing computational, storage and resource provision services in a multi-tenant environment, saving process and people cost [1]. Cloud Computing offers different models and services to different tenants based on their compute requirements. Industry 4.0 (I4.0) which is based on the new technologies and communication models (i.e. Advanced robotics, Artificial Intelligent, Big data, Industrial-IoT, 5G, etc.) highly relies on cloud computing for its data processing requirements. Data security is the key factor driving the I4.0 environment, data analysis helps in making efficient decisions related to the production, transparency, product work flow and flexibility. Since 2007, the Industrial Control Systems have suffered cybersecurity breaches (Malware, Keyloggers, Denial of Service, Tampering, etc), Stuxnet, German Steel Mill, Ukraine Power Grid[1], are just a name to few. Cloud computing offers different applications and services, many of which are provided by third-party subcontractors. Based on ENISA's report, a wide range of vulnerabilities arise when services are provided by sub-contractors[2]. Cloud vendors state that these issues can be controlled through a Service Level Agreement but majority of the SLA contracts are improperly defined, ambiguous and end-in vendor lock-in situation, which are discussed in this paper in detail.

Cloud Service Level Agreement (CSLA) measures and monitors the performance and Quality of Service (QoS) promised by the vendor [2]. A CSLA is based on several parameters such as: response times, identity and access management, availability, versioning, etc. Each of these prime

[1] https://teskalabs.com/blog/industrial-iot-it-ot-convergence

[2] https://www.enisa.europa.eu/publications/enisa-threat-landscape-for-5g-networks

Lubna Luxmi Dhirani and Thomas Newe, "Hybrid Cloud SLAs for Industry 4.0: Bridging the Gap", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, pp. 41-60, Vol. 4, No. 5, 20th December 2020, Published by International Association of Educators and Researchers (IAER), DOI: 10.33166/AETiC.2020.05.003, Available: http://aetic.theiaer.org/archive/v4/v4n5/p3.html.

parameters may depend on sub-metrics for precisely monitoring the services. Since cloud vendors promise services in percentages (i.e. 99.95%-99.99%), services dropped by 00.01 percent may also be counted as a service violation. These parameters and sub-metrics are the source of measuring the CSLA QoS. Vendors may also have pre-defined exclusions to a CSLA, such as: hardware, software, network or monitoring failure, scheduled downtime, Denial of Service (DoS), force majeure, etc. These service disruptions may highly impact I4.0 tenants if their hybrid cloud is configured to operate on a single geographic zone and lead to major operational and financial loss. Some cloud vendors provide tenant applications to run on dual zones, just to overcome single zone failure/unavailability issues. This feature may not be implemented by I4.0 tenants as they are bound to Government rules and policies (i.e. Data Governance Risk and Control (DGRC)) where data processing is limited to certain jurisdictions, leaving the tenant bound to a single zone and availability/outage issues. Though the tenant will suffer a downtime (operational disruption) based on the type of SLA (i.e. 99.95%, 99.99%, etc.), it will still be considered valid and within the SLA at the vendors end as it is generally mentioned as an exclusion in the I4.0 tenant-vendor CSLA.

Service Level Agreement offerings may differ for different cloud models (i.e. public, private, hybrid and community) and architectures (i.e. Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS)) [3]. Each application running under the following architecture (SaaS, PaaS and IaaS) comprises independent SLA with different parameters, availability set and exclusions. Public SLAs are generalized and easy to understand as the tenant's applications are running in the public cloud only. Private cloud applications are on-premises and SLAs managed by organizations themselves, through IT Operational Tools providing insights, visibility and control of the environment. These SLAs become critical when organizations form a hybrid cloud, since some of the applications are moved on a public cloud or run under both public/private simultaneously and are subject to different QoS metrics. Community clouds work in a multi-tenant environment infrastructure but with organizations which share common goals such as: different government departments may share security, jurisdiction or compliance-based data among each other. Community clouds may be on-premises or off premises, the CSLAs may share commonality as the departments may use the same application for holding confidential public data. Hybrid cloud SLAs involve complexity as applications are being processed on both public and private domains, tenants majorly put non-critical applications to be processed in hybrid environments, but they also contribute to the organizations overall support and functionality.

This paper reviews different public cloud vendor offerings hybrid cloud setups, highlighting the existing SLA shortcomings in the hybrid cloud environment IaaS architecture and to create a better understanding for the Industry 4.0 tenants and non-experts. The paper is structured as follows: Section 2 covers SLA introduction, policy, pricing, functionality and dependencies. Section 3 compares well-known vendor services and SLAs. Section 4 demonstrates SLA variations for cloud architectures in context of an industrial cloud tenant, Section 5 highlights SLA limitations in the hybrid environment, Section 6 discusses Cloud SLA legal and privacy issues for Industry 4.0 and viable solutions, Section 7 opens a room for discussion and Section 8 concludes the paper.

2. What is a Service Level Agreement?

Outsourcing complex IT infrastructure to public cloud vendors has rose to fame and led to much recent development due to cost associated factors. To ensure QoS between a tenant and the cloud vendor, they mutually outline a Service Level Agreement (SLA) as a part of a cloud service contract that can be supervised by whichever party or a third party [4]. An SLA provides metrics for measuring the Service Level Objectives (SLOs) performance levels. SLA guidelines are used to assess the cloud service implementations and detect SLO violations. SLAs are essential for any category of IT-based outsourced processes and assume a dominant place in IT Service Management (ITSM) standards such as ITIL. IT Service providers process thousands of SLAs per day for different tenants and distinct types of services in the service-oriented computing landscape. Many commercial Service Level Management (SLM) tools exist such as: Microsoft Operational Management Suite, IBM ITOM, Ansible, etc. [3]. These SLM save selected QoS attributes such as:

uptime or availability as parameters in the application code or database tier. However, this method is limited to static rules and insufficient set of parameters [4]. In [5] many text-only real world SLAs are analyzed and the authors found distinct types of policies which need to be imposed by IT service providers.

2.1. SLA Attributes

SLA attributes may vary from vendor to vendor or may use different names. SLA attributes may possess discrete characteristics and features broadening the scope for cloud ecosystem. Cloud models SLA QoS parameters may also vary for different cloud models (i.e. IaaS, PaaS and SaaS) as they differ in terms of applications and service which is discussed in detail comprehensively in this paper. Table 1 depicts a list of generalized SLA parameters.

Table 1. Generalized SLA Parameters [6]

QoS Metrics	
Availability Metrics	Availability metric rate (i.e. 99.5%), Downtime/Week, Downtime/Month, Downtime/Year, Outage Duration.
Reliability Metrics	Mean-Time Between Failures, Reliability Rate
Performance Metrics	Network Capacity, Storage Device Capacity, Server Capacity, Web Application Capacity, VM Starting Time, Uptime, Accomplishment Time
Scalability Metrics	Storage Scalability (Horizontal), Server Scalability (Horizontal/Vertical)
Resilience Metrics	Mean-Time to Switch Over, Mean Time System Recovery.

Each parameter may comprise SLA details such as: SLA description, measurement, method and frequency of collection, threshold levels, cloud delivery model and availability for evaluating the SLA QoS.

2.2. SLA functionality and dependencies

SLA metrics often share functionality and performance dependencies. Figure. 1 shows the authors designed illustration for service quality metrics called Instance Starting Time Metric (ISTM). The role of this instance is to measure the instance performance based on the frequency, pre-defined vendor threshold, date and time details for instance requested and the duration it took to respond to the tenant requests [6]. Each SLA metric triggers other metrics and sub-metrics such as: average response time, throughput, billing cycles, service credits, etc. These metrics have a pre-defined threshold, response time and measurement as well.

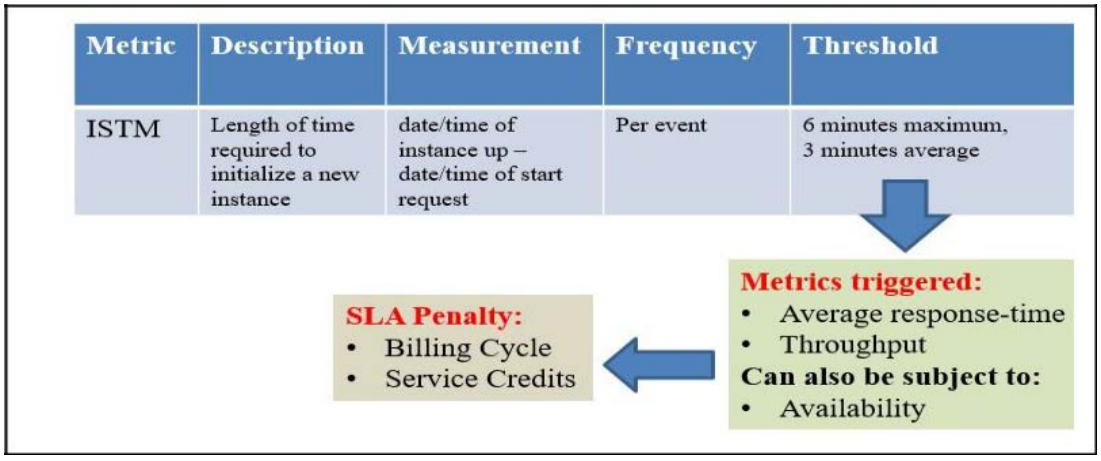

Figure 1. Metrics dependencies

As per [7] cloud risks can be mitigated for vulnerability with SLA guarantees with webservices providing the following attributes, such as: Statefulness, Access, Response-time, Time-out and Versioning. Though, it may exempt potential threats from the SLA such as: hardware, software performance monitors and network failures, Denial of Service, acts of God, scheduled upgrades and backups, etc. Vendors may differentiate with naming the SLA attributes, for example: Elasticity (cloud metrics) may be used by vendors instead of resource provisioning, scalability and agility metrics. For Example: the cloud metrics (i.e. throughput, response time for initiating a new instance

on the IaaS, user-threshold levels, data requests threshold level, resources threshold levels) have a significant impact on the tenant's hybrid cloud performance but majority of vendors do not provide reliable QoS metrics [6] creating confusion and service issues in the hybrid cloud environment.

2.3. SLA rules

Service Level Agreements (SLAs) have become highly substantial, as they define policies for the provisioning and implementation of cloud services including the security aspects. The essential Cloud SLAs security requirements, include asset sensitivity, legal, governing policies and cloud providers security competences. This includes the privacy rules and regulations based in different regions in which cloud vendors may store or process the tenant's data. The authors discussed the Data Governance Risk and Control (DGRC) issues in Section 4.3 in detail.

These SLA rules combine the technical, organizational and legal components together. For example, QoS parameter service availability (technical component) falling less than the promised threshold level may activate the SLA monitoring tool (organization component) to report this issue to the cloud vendor (legal component) for violating the promised services.

Like other basic utilities, cloud vendors may also differentiate charges based on prime, peak or standard timings for the resources granted [4]. However, just like terms and conditions associated with IT based services, in a cloud environment the attributes stated in exemptions are automatically voided from the SLA penalty. SLA rules depict the feasibility of the SLA attributes, since tenant requirements may change or vary weekly, monthly or yearly therefore it is compulsory for SLAs to update automatically and be of a dynamic nature in assuring the SLA QoS.

As discussed in Table 1, each SLA may be linked with another set of attributes and needs to follow a relational formalization to calculate the parameter thresholds correctly. The complexity of an SLA increases in hybrid and multi-cloud environments which is discussed in section 4.

2.4. SLA categorization

In [4], the authors categorize SLAs as follows: (i) Basic agreement (ii) SLA (i.e. Promised QoS guaranteed 99.5%, 99.95%, 99.99%) (iii) Group SLA as shown in author's illustration (see Figure 1) and Table 1 (a single parameter invoking another set of QoS attributes which must be capable of functioning together) (iv) Operation Level Agreement (Service Agreement with internal partners for operational level guarantees such as network or software providers) (v) Underpinning Contract (i.e. Shareholders).

The nature of SLAs dynamically changes based on tenants demands for computation and resources. Each SLA may be linked with another group of SLAs which need to be met to fulfil the QoS parameter. The basic agreement conceals the overall outline and framework for the cloud services whereas the Operational Level Agreement focuses on the contractual SLA requirements to be met by the internal operational partners. The underpinning contract is based on contracts with external operational partners, if the I4.0 tenant has any this may apply.

2.5. SLA pricing policy

The SLA pricing policy [9] is the most critical and confusing characteristic for new cloud tenants. As mentioned before pricing may vary as per peak, prime or standards timings. Another categorization for pricing is: fine-grained or coarse-grained [9].

- Fine-grained pricing may be based on per minute or per second charges [10].
- Coarse-grained pricing is based on hourly basis, whether the resources are fully utilized or not the tenant is liable to pay for the full hour.

Previously, well-known vendors offered only coarse-grained pricing offers but with recent competition vendors such as: AWS and Google Cloud Platform have initiated the fine-grained per minute pricing to maintain their monopoly in the cloud environment.

AWS offers per second billing for AWS Elastic Cloud Compute Engine (EC2) and Elastic Block Storage (EBS). Google follows the same trend for its cloud offerings. Fine-grained billing may only be feasible for tenants with less computational needs or tenants running short-run batch jobs at off-

peak hours. Tenants who have already bought their computational needs in advance (3-5 years) may not find the fine-grained pricing attractive as they generally need computational resources round the clock[3].

Based on tenant's contracts, services are provisioned. For example, a flexible tenant contract may grant a leverage for requesting more resources as the computational demand increases, unlike fixed contracts where additional resources may not be granted or may be subject to availability. Cloud Instances offered by cloud vendors comprise of the following features: operating system, computation, memory, instance, storage type, storage region and category of the contract[3] as shown in Table. 2. AWS[4] further categorizes its instance types as following:

- On-demand instances follow the pay-as-you go strategy without any prerequisite condition.
- Reserved instances are only granted when a prerequisite for a long-term con-tract and upfront consumption costs is approved.
- Spot instance function based on bids and on-spot prices which varies based on available instances. Spot bidding provides cost benefits since workloads can be ended anytime whereas spot-blocks are subject to fixed interval, it does not intervene in terminating the instance midway, but the processing may finish when the allotted time-period is over.

Table 2. Vendor Instance Types[5,6,7]

	AWS (EC2 and EBS)	**Microsoft Azure**	**Google Cloud Platform**	**IBM**
Type	General purpose, Compute optimised, Memory optimised, Accelerated Computing Storage optimised Burstable performance and Elastic GPU [9]	General Purpose, Compute optimised Memory optimised Storage optimised GPU, High Performance Compute	Shared Core Standards, High Memory, GPU. SSD Storage	Bare metal servers Virtual servers (Public, Transient, dedicated)
Billing	Per hour/Per minute (Subject to SLA contract	Per minute	Per minute and after the first minute the tenants are charged on second basis	Lite (free-limited), pay as you go subscription.
Pricing offers	Reserved, on-demand and on-spot.	Enterprise agreements, Azure Dev/test pricing, Enterprise agreement support offer	Spot blocks, sustained use discounts, committed use discounts, Pre-emptive VM Instances	Monthly bare-metal, hourly bare-metal, hourly public and dedicated servers
Instance Offers		Microsoft Azure, Hybrid benefit, Azure Government customers.	Instance customisation offered. Instance right sizing recommendations.	Negotiable offerings.
Example of Instance Equivalence	t2.micro – t2.large c5.large	Standard_B2s Standard_B2ms (Approximately near) F2v2	Shared Core f1-micro g1-small n1-highcpu-2	- C1.2 x 2.25

Cloud computing deployment costs add up with the following services: Premium CPU Platforms, Red Hat Enterprise Edition Images, SQL Server Images, network pricing (i.e. ingress, egress), load balancing (i.e. ingress and egress charges) and protocol forwarding, traffic through external IP address, Virtual Private Network (VPN), disk pricing, unused IP address pricing, etc.

[3] http://itknowledgeexchange.techtarget.com/cloud-computing-enterprise/is-per-minute-billing-the-next-step-to-unlimited-cloud-plans/
[4] http://blog.armory.io/choosing-between-aws-gcp-and-azure/
[5] https://aws.amazon.com
[6] https://azure.microsoft.com/en-us/
[7] https://cloud.google.com

Cloud vendors exempt the pricing offers for small instances and length of contract. Therefore, it is necessary for I4.0 tenants to evaluate the costs from all aspects before adopting the cloud platform. Pricing schemes, Virtual Machines (VMs) and instance types have been discussed in detail in [9]. Cloud vendors may change the cloud pricing schemes as any time giving a short notice period to the I4.0 cloud tenants to make a decision, leaving them in a vendor lock-in type situation.

2.6. SLA management for Public, private and Hybrid Cloud

Public cloud SLAs depicted on cloud vendors website are generic and may not show the technical aspects of the cloud and SLA management. For example: AWS SLA which was last updated on February 12, 2018 [13] states "Service commitment definitions associated to availability zone, monthly uptime per-centage, region unavailability, service credits and commitments, payment methods and SLA exclusions only". There is absolutely no detail provided of how the SLA will be managed when services are subcontracted to third-party subcontractors. Public SLAs may only be monitored by the provided vendor's graphical user interface cloud portfolio. If tenants require additional security they may have to purchase additional packages in AWS (i.e. AWS Inspector, AWS CloudWatch AWS CloudTrail)[8], Microsoft Azure (Cloudyn and Azure Monitor) and Google (Google SlackDriver)[9]. These additional services assist in services managing and monitoring, but they do increase cloud management costs too. For example: The AWS Inspector[5] pricing is based on agent-assessment and is calculated as following:

No. of assessment runs x No. of agents or systems assessed during those runs = agent assessment.

Where α represents the no. of assessments runs, β represents the no. of agents or systems assessed during those runs and γ represents the agent assessments.

$$\alpha \times \beta = \gamma \quad (1)$$

An on-demand billing period is one calendar month like all AWS services. The pricing of each individual agent-assessment is based on a tiered pricing model. The more volume of agent-assessments in each billing period the lower a tenant may pay, but it is still an additional cost.

Private clouds are based on-premises and may not be subject to unavailability, as there are alternative storage and network paths defined to keep the IT operations running in IT failover situations. Private clouds may only monitor SLA for applications (i.e. ERP, DSS, etc.) bought by software houses for supporting their core business performance.

The hybrid cloud is a fusion of private and public cloud services with communications between the platforms for IT operation management. This model provides a business flexibility to focus on their core business and different deployment models. Aside from the benefits, hybrid cloud model may present technical, business and cloud management challenges. Hybrid cloud SLAs need to be managed effectively. Previously I4.0 tenants were only running their non-critical business processes on the public cloud and keeping their critical data and applications on the private cloud but currently tenants prefer running applications on the hybrid environment due to latest IoT based services provided on the public cloud. Industrial applications running on the hybrid cloud must be designed to function in sync, be compatible and fully integrated, to avoid complexity.

3. Vendor Service Level Comparison

This section differentiates and highlights shortcomings in different vendor SLAs.

3.1. Vendor Service Level Agreement

SLA Metrics are parameters for assessing the SLAs performance levels based on a variety of parameters, such as: computing, storage, scalability, availability, etc. Table 3 highlights the types of service offered by well-known cloud vendors, Amazon Web Service (AWS) and Google Cloud

[8] https://aws.amazon.com
[9] https://cloud.google.com

Platform (GCP) aggressively and do have options for I4.0 tenants who wish to switch from AWS to GCP or GCP to AWS[10]. The vendor services for each architecture (IaaS, PaaS and SaaS) also vary based on their expertise and global reach to third-party subcontractors. AWS is more IaaS focused where as Microsoft Azure concentrates on the PaaS and SaaS architecture. Google Compute Platform (GCP) inclines to support start-ups and focuses on SaaS and lastly IBM build its own services and has contributes 50-75% in SaaS, PaaS and IaaS architectures.

Table 3. Well-known Public Cloud Vendors Summary Chart

Cloud Vendors	Build Private Services	Deliver Services	Services Delivered			Private Offerings	
			IaaS	PaaS	SaaS	Enabling Technology	Packaged Cloud
AWS	0	5	5	2	3	2	0
Microsoft Azure	5	5	4	5	5	5	5
GCP	0	5	3	4	5	1	0
IBM Cloud	5	3	4	3	3	5	5
Significance (min-max): 1 (min) – 5 (max)							

Amazon Web Services (AWS). AWS's generic SLA has already been discussed in Section 2.6, however each product or service AWS offers is subject to a separate set of SLAs in terms of performance, availability, exclusions and upgrades. AWS offers a wide variety of instances, applications, services and serverless computing and integrates seamlessly with other Amazon services. AWS being the pioneers in the cloud sector have been the longest in this industry compared to the others which means that a lot of sub-services and tools for cloud deployments all support AWS integrations[4]. Amazon EC2 service exclusions: *"The Service Assurance does not apply to downtime, interruption or service closure of EC2/EBS performance issues in terms of force majeure, sub-contractor service downtime, region inaccessibility, software or hardware failure, etc. If availability is clogged by issues different from the those listed in the monthly uptime percentage calculation, then AWS could grant a Service Credit considering such factors on the vendors choice"*[5]. Majority of the service violations are simply exempted by AWS by placing them in the SLA exclusions, which means that I4.0 tenants may not be eligible for service credits despite of experiencing an outage. Data confidentiality, integrity and availability are essential for I4.0 at the IT/OT level. Failure to comply with these requirements may lead to complete system failure, loss of control, product defect, financial or human loss.

Microsoft Azure. *"Microsoft Azure's SLA describes Microsoft's provides assurance for uptime and connectivity only"*[6]. The SLA for individual Azure services are listed explicitly which means each service (i.e. compute, networking, storage, security, monitoring, etc.) has a different availability, upgrade, region and exclusions. Tenants moving their applications on to the vendors cloud may have to track multiple SLAs since each service may be subject to unavailability at a different time leading the tenant's IT operations to halt. Recently, a load-balancing fault lead Hotmail to 12 hours of downtime in the UK and European region, despite having fault and network management configurations, cloud vendors may suffer unplanned outages [12]. Microsoft Azure offers a single-instance VM SLA of 99.9% on VMs backed by Azure Premium Storage and 99.95% SLA for availability sets[6]. An Availability Set denotes to multiple VMs installed across diverse Fault Domains avoiding lone failovers. Azure offers smaller and flexible server configurations. VMs perform better providing flexibility as compared to AWS but is limited in terms of VM Instance types. Alike AWS, Azure stores data in Blobs but is offered at different SLA levels such as[4]: Locally redundant storage (LRS), Zone redundant storage (ZRS), Geographically redundant storage (GRS) and Read-Access Geographically Redundant Storage (RA-GRS). Azure offers multiple availability zones, greater resilience and Databox which gives the tenants mail the flexibility of up to 100 TB of data from private datacentres to the cloud. Azure would be a smart choice for I4.0 tenants requiring workload with low latency, a range of datacentres, and comprehensive VM settings. The pricing structure[11] is based on either pay-as-you-go instances or advance payments for reserved instances which may offer discounts at times as well.

[10] https://www.forbes.com/sites/louiscolumbus/2013/02/20/de-mystifying-cloud-vendors/#4ca0537f40f7

[11] https://docs.microsoft.com/en-us/azure/virtual-machines/windows/prepay-reserved-vm-instances

Google Cloud Platform (GCP). GCPs SLA describes different service levels for each of its application provided to the tenants. The GCP terms of service clause covers the GCP license agreement as follows[7]: provision of services, billing terms, Intellectual Property Rights, use of tenant data, termination clauses, assurances, exclusion, etc. Google supports customized and assorted instances and provides control over tenant-defined machine/instance settings, but complications may arise while running N-amounts of machine/instance types[4]. GCPs sustained-use pricing for compute services provides flexible approach to AWS's reserved instances[7]. Based on tenant's continuous consumption, the on-demand baseline hourly rate automatically implements the discount on the tenants' account which means that there are no upfront requirements for concessions. GCP overcomes the latency issues with its global network infrastructure that Google controls, but this may require the tenant's workload to be processed in region specific datacentre[7]. As per GCP Terms of Service clause 12, disclaimer states *"Google and its sub contactors do not guarantee any sort of assurances of merchantability, capability for a precise use and non-infringement. GCP and its sub-contractors exempt responsibility for tenant's data processing, transfer and backup and do not authenticate that the software's/services operating on the cloud will be uninterrupted"* [12]. Based on the above, GCP does not take responsibility in terms of service failures, force majeure, services which are out of their control or service denials that may take place due to a cloud broker or third-party sub-contractor leading the tenant vulnerable to security and DGRC, such requirements are essential for I4.0 tenants.

International Business Machines Clouds (IBM Cloud). *"IBM[12] does not assure an error-free, continuous cloud service operations or avoid third-party disruptions or unauthorized third-party access. These guarantees fall under the exclusions such a misuse of services, damages, modification, non-infringement, etc.".* Considering the above-mentioned SLAs, the I4.0 tenants would be susceptible to a number of security-based risks.

The cloud SLAs published by AWS, Azure, GCP and IBM, make it evident that none of these vendors guarantee the cloud QoS, processed on-premises or via a third-party subcontractor leading the tenant susceptible to operational and security-based risks. Table. 4 illustrates service comparisons between AWS, Azure, GCP excluding IBM, as IBM does not provide its service details publicly.

Table 4. Well-known cloud vendor service comparisons

AWS[5]	AWS EC2
GUI	AWS Management Console
Instance Families[13]	7
Instance Types	38 (Regions, Zones)
Storage	Amazon S3
	$0.03 (Standard)/$0.0125 (Infrequent)[14]
Archive (Cold Storage)	Glacier
	$0.004
Data Storage	Offered in four different SLA Levels. Amazon S3 stores data in multiple datacenters and each datacenter is linked with multiple storage devices.
Network and I/O Disk	Failure in Network Attached Disk may lead to fail-over and switching to another availability zone or region. EC2 Metric for Network: NetworkIn, NetworkOut EC2 Metric for Storage: DiskReadOps. DiskWriteOps
Compute	AWS EC2 VMs based on KVM for C5 instances and Xen hypervisor for other instances.
Disk	Disk Solutions: EBS C4/R4/M4 instances. The I3/D2/X1 series have exceptional choices for local disk. Availability depending on the Disk type may vary between 99%-99.9%
Support	Fair [15]
Database	Amazon RDS

[12] https://www-03.ibm.com/soft-ware/sla/sladb.nsf/pdf/6388-02/$file/i126-6388-02_03-2014_en_US.pdf
[13] www.ibis.in.tum.de
[14] https://tecsynt.com/blog/research-and development/cloud-platform-comparison-for-2017
[15] https://metamar-kets.com/2017/big-cloud-data-aws-and-gcp/

Network	The instances give general provisions for low/medium/high network limits. The SLA of the GBs instances, only mentions throughputs (using placement groups), which may lead to freezeouts where tenants may not get capacity which means that the tenants network throughput and consistency will be highly variable and unpredictable.
Availability Zones (AZ)	Redundant VMs AmazonRDS: 16AZ AWS EC2: 14AZ
Billing	Billing Dashboard: AWS shuffles around zones/account which makes tracking complex with respect to zone correspondence. The billing is stated by spot-pricing sometimes whereas invoicing documents fail to refer to specific zone alphabetical notation. The cloud expense is accumulated as line entry where dissimilar line items have rate pointers. These rates are multiplied by the consumed resource quantities following a predictable denormalized data scheme that is compatible with multiple analysis tools[13].
Cost	Considered cost friendly and competitive, but the pricing and offering lead are quite confusing, on the other hand the bill lack clarity of how data is being processed in another region than defined and billed.
Security	AWS Key Management Service (Data Encryption), AWS Inspector, AWS Config, AWS CloudFormation (Inventory and Configuration), AWS Shield (DDoS Protection), AWS IAM, AWS VPC (Virtual Private Connection), etc.
Monitoring	AWS CloudTrail, AWS CloudWatch (Monitoring and Logging).
Other Features	The maximum memory/core available is higher in AWS than GCP (unless the I4.0 tenant pays a premium)
Azure[6]	AzureVM
GUI	Azure Portal
Instance Families[11]	4
Instance Types	33 (Regions, No zones)
Storage	Microsoft Azure Storage
	Microsoft Azure (Data Lake Store) $0.04
Archive (Cold Storage)	Azure Backup
	$0.01 (LRS)/ 0.02 (GRS)/$0.025 (RA-GRS)
Data Storage	Data storage is done in Blobs, like AWS it is implemented in four different SLA levels: LRS, ZRS and GRS and RA-GRS[4].
Network and I/O Disk	Azure VMs have at least two disks: a Windows OS disk and a provisional disk. A data disk is a VHD that is attached to a VM to store application data or other data.
Compute	AzureVM is Hypervisor-V. Azure supports KVM for Dv3 and Ev3 Series as well.
Disk	Azure Disks support 99.999% uptime allowing I4.0 tenants to have multiple data replications enabling high tolerance against failures. This architecture provisions Azure to continuously deliver enterprise-grade robustness for IaaS disks. The tenant data is routinely encrypted at rest which might be of concern for the I4.0 tenants data security.
Support	Good
Database	Azure SQL, Document DB
Network	Bandwidth denotes data moving inbound and outbound the Azure datacenters. The highest bandwidth provides direction for I4.0 tenants to choose the correct VM type while ensuring the network capacity. While shifting between different threshold levels (i.e. low/moderate/high/highest) the throughput upsurges consequently. The authentic network performance depends on varied factors such as: network, application loads, etc.
Availability Zones (AZ)	VMotion
	36 Regions globally [16]
Billing	It is done based on the subscription chosen. The billing metrics includes: the billing cycle, name (meter category), Type (meter sub-category, Resource (meter name), Region, Consumed, Included Quantity, Billable (Overage Quantity)
Cost	Azure cost management creates complexity for I4.0 tenants, since resource need to be monitored continuously, until the stop metrics is defined.
Security	Security and privacy features are designed in the Azure platform. Other security service may include: Azure and Data Encryption, Key Vault Service, Azure DDoS Protection, Azure Fabric Controller, Network Security Groups, Azure Security Centre, etc.
Monitoring	Azure Monitor, Cloudyn (Monitoring and Logging)
Other Features	Azure's Hybrid Cloud is competitive and is expanding its regional growth by deploying new datacenters.

[16] http://map.buildazure.com/

GCP[7]	GCE
GUI	Google Cloud Console
Instance Families[11]	4
Instance Types	18 (Regions, Zones)
Storage	Google Cloud Storage
	$0.026 (Standard)/$0.02 (DRA)[12]
Archive (Cold Storage)	Cloud Storage Nearline
	$0.01 (Storage) + $ 0.01 (Retrieval)
Data Storage	*"It has the smallest infrastructure, with four regions, comprised of 3-4 "zones" (data centers) each. Other data centers provide regional support against zonal failures and act as redundancy only"[4].*
Network and I/O Disk	*"Persistent disks are the common storage options due to their price, performance, and predictability. Instances can be created with local SSDs for better performance, low latency, but without the data redundancy and robustness. Block storage performance comparison: GCP provides finding the correct storage size and performance requirements for selecting the correct disk type and size for tenant's instances. Performance requirements for a given application are typically separated into two distinct IO patterns (i.e. small reads and writes, large reads and writes)[7]".*
Compute	GCP GCE VMs is KVM
Disk	Disk resources can only be accessed by other instances within the same zone. Disks attached in one zone as an instance cannot be attached to other zones. The Standard persistent disks, SSD persistent disks, Local SSDs, cloud storage buckets are supported by all machine types and available at regional zones but the availability % is not mentioned. GCE encrypts tenant's data before it moves to the persistent disk storage space and these disks stay encrypted with system-defined keys/tenant-supplied keys.
Support[13]	Good
Database	Cloud SQL
Network	Networking per VM is consistent and higher than AWS. The achievable network capacity is based on the number of CPUs tenant's virtual machines has. GCP provides the flexibility for defining specifications and delivering anticipated throughputs but does not provide fixed bandwidth for network attached storage but has higher network availability.
Availability Zones (AZ)	Persistent. Zone failures managed by load balancing and regional/zone diversity
	13 regions globally[7]
Billing	Approximations are provided based on which the tenants can build Data Studio reports[12]. From a network logistics standpoint, GCP leads in zone labelling and provides same zone names for all tenants. The billing is exported into BigQuery, each line entry is a combined over a span of accrued usage and have sub-components of credits. Their current billing has the following flaws (i) auditing is very hard, the tenant must take the numbers as presented, calculating the figures independently is hard. (ii) calculating estimated spend is complex, accumulating extra jobs for the financial teams[12].
Cost	Cost effective cloud for certain applications and for new cloud (tenants), as billing is based on fine-grained criteria.
Security	GCP Cloud IAM, Cloud Platform Security Overview, GCP DoS Protection, GCP VPC, GCP Physical Infrastructure Security, etc.
Monitoring	Google StackDriver (monitoring, logging, and diagnostics for GCP and AWS apps).
Other Features	The GCP offering of custom machine types. AWS EC2 and GCP both supports networked and locally attached block storage.
Commonality	
AWS, Azure and GCP support containers	
Cloud Failures: AWS, Azure and GCP have their own set of limitations and all have undergone outages and failures, and therefore there are so many exclusions mentioned in Vendor SLAs.	
Threats: Hardware failure, software failure, acts of God, network issues not in direct control on the vendor, scheduled upgrades, indemnification, force majeure, misconduct of service by a third-party or tenant, scheduled upgrades, etc. AWS target attacks are quoted to be in hundreds whereas GCP is subject to more than 100,000 attacks per day[13].	

Cloud computing is a heterogenous environment forming new models and architectures, each diversifying in services, application and performance. The conventional cloud models are unfit in terms of the current computational demands and architectures because of:

(i) Lack of standardization in the SLA specifications and parameters leads to confusion in a multi-cloud hybrid environment
(ii) Customising issues.
(iii) Cloud heterogenous models require visibility, insights, flexibility, reliability, compatibility and platform independence for monitoring and managing the hybrid, multi-cloud and federated cloud deployments.

4. SLA Variations for Cloud Architectures

This section discusses the SLA complexity with respect to different cloud architectures. Cloud computing offers three architectures: IaaS, PaaS and SaaS and each of these architectures are subject to different availability and QoS parameters. The authors focus on mainly the IaaS architecture through the rest of the sections as Industrial sectors mainly rent the computational and infrastructure-based (i.e. servers, storage, applications, etc.) services from cloud vendors. Although the tenant pays for all the resources used on the cloud vendor's infrastructure, the vendor only holds responsibility for the virtualization, servers, storage and networking layers which may also be subject to the SLA uptime and availability. PaaS is renting the cloud platform for building applications and is usually implemented by programmers who do not wish to invest in platforms as renting the resources has a cost benefit and ease of access to the services. The cloud vendors hold responsibility for all layers apart from the application and data management. SaaS is using Software as a Service. For example: Enterprise Resource Planning (ERP) Software, Microsoft 365 online, etc. Since it is a consumer application and provided by the vendor, therefore it holds responsibility for all layers. Any packaged or third-party software used by I4.0 cloud tenants which may run on the cloud vendor's infrastructure may solely be the tenant's responsibility. Table 5. summarizes the well-known cloud vendors architecture offerings[7,17] and liability[18]

Table 5. AWS, Microsoft Azure, Google And IBM Cloud Model Offerings[16]

Computing component	SaaS: Software as a Service	PaaS: Platform as a Service	IaaS: Infrastructure as a Service	Packaged Software
Usage	Consumer App	Build App	Host App	Tenant Managed
Application	Vendor handled	Self-governed	Self-governed	Self-governed
Data Management	Vendor handled	Self-governed	Self-governed	Self-governed
Runtime	Vendor handled	Vendor handled	Self-governed	Self-governed
Middleware	Vendor handled	Vendor handled	Self-governed	Self-governed
OS	Vendor handled	Vendor handled	Self-governed	Self-governed
Virtualization	Vendor handled	Vendor handled	Vendor handled	Self-governed
Servers	Vendor handled	Vendor handled	Vendor handled	Self-governed
Storage	Vendor handled	Vendor handled	Vendor handled	Self-governed
Networking	Vendor handled	Vendor handled	Vendor handled	Self-governed

4.1. IaaS Issues

This section briefs the IaaS architecture, Hybrid multi-tenancy cloud challenges, database scalability, web services and legal issues associated to Service Level Agreements QoS.

Since AWS, Microsoft Azure and GCP do not publish their architecture blueprints the authors continue this section with IBM cloud providers SLA monitoring and management model (i.e. IBM multi-tenant cloud IaaS) as shown in Figure. 2. The authors believe that vendors AWS, Azure and GCP may implement a similar IaaS model like IBM. IBMs cloud infrastructure consists of cloud catalogue, virtualization, infrastructure, storage layers, etc. These layers need to be aligned with security and compliance measures.

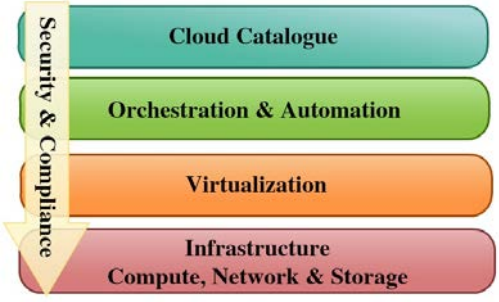

Figure 2. Adapted from IBMs existing Cloud model[19]

[17] https://bi-guru.wordpress.com/tag/aws/

[18] https://stack247.word-press.com/2015/05/21/azure-on-premises-vs-iaas-vs-paas-vs-saas/

[19] www.ibm.com

4.2. Hybrid multi-tenant, multi-cloud IaaS model

A Hybrid Cloud is a fusion of a public and private cloud model. Cloud vendors encourage multi-tenancy to fully optimize their virtual resources and capacity. Since multiple tenants share the same virtual infrastructure it is susceptible to various risks such as: hacking, data theft, identity and access management (IAM) and Denial of Service (DoS), etc. which is a major security issue for I4.0 tenants.

Existing I4.0 tenants which run their applications on their existing private cloud, need to customise and integrate applications, considering the risks it may pose before migrating them on a hybrid cloud (multi-tenant environment). Migrating private clouds on a vendor's environment may require complying with the cloud standards and security measures (as shown in figure 2). The authors emphasize, that situations in which an Industrial tenant moves moves its multiple private clouds processing to the hybrid multi-tenant environment, security and compliance of the applications and services need to be orchestrated and automated across the IaaS model. The authors have designed a model of their recommendation as shown in figure 3.

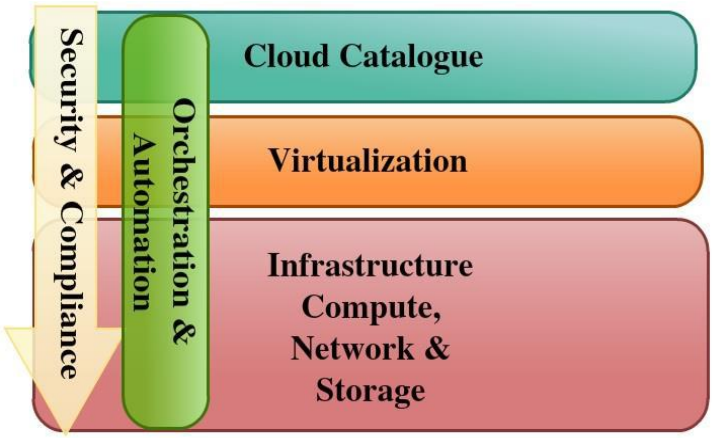

Figure 3. Hybrid multi-tenant, multi-cloud IaaS model (author's recommended model)

4.3. Hybrid cloud multi-tenancy challenges for Industry 4.0 tenants

This sub-section highlights the challenges associated to a hybrid cloud multi-tenancy. Multi-tenancy requires I4.0 cloud tenant's integration at the following levels: (i) Data Centre (DC) layer (ii) Application (iii) Infrastructure layer [13].

(i) Data Centre (DC) services are provided by a cloud vendor or further processed with a sub-contractor hosting network, storage and computational resources within the same regional premises configuring security measures such as: firewalls and access controls to prevent tenant's applications and services from malicious attacks in the compute shared environment.

(ii) Multi-tenancy at application layer is challenging since tenant applications require modifications/redesigning of the existing software at the architecture level [13]. It also adds complexity while migrating/processing Industrial legacy applications on the hybrid cloud as the applications were not initially designed to function on a hybrid environment. Programming languages used for previous generation of software's had limitations in terms of scalability, reliability and were often hardware specific. Cloud vendors provide access to I4.0 tenants via web services or Hyper Text Transfer protocol (HTTP) interfaces which act as pointers to directly access the tenant's applications. Vendors differentiate the tenants by linking their account details in the HTTP Uniform Resource Locator (URL) making the process simpler for tenant requests [13].

Application servers have a significant impact on the application layer as it modifies the way applications are installed and configured [13]. Tenants anticipate the application servers to provide high performance and scalability during peak processing hours. The

application code changes are required as multiple tenants are running their applications on the same server which may lead to lower or higher response times depending on the number of transactions being processed. As mentioned before, the I4.0 tenants application security is a matter of concern as it shares the same application server and logical system memory with the other cloud tenants on the VM [13].

(iii) Many cloud vendors pre-configure the Infrastructure layer making it easier for physical or virtual resource deployment, since it exempts the need for application code modifications until and unless the applications have specific requirements [13]. Insecure Application Programming Interfaces (APIs) [14] may lead the I4.0 cloud tenants susceptible to authentication, authorization, access control and monitoring of the application during runtime, leading to a high-level security breach.

Cloud servers capable of hosting multiple VMs allows multiple applications from multiple tenants running under the same VM making it vulnerable to malicious attacks. The complexity lies in tracing the security incident as information is traced out due to large volumes of write operations on the storage media [13]. Database scalability requires to be aligned with application and infrastructure layers. Multi-tenancy is implemented at the application-layer and may need to adhere the database schema patterns which apply to multi-tenant architectures. Multi-tenant databases issues can only be prevented if the tenant's applications are designed considering the database sizing and following the vendor's recommended database table spaces. Since database scalability is implemented on two different layers the SLA availability set may vary and the overall performance may from 99.95% to 99.5%, as the SLA is assessed cumulatively.

Resource management needs to be planned at the network level as well which is subject to a different SLA. Cloud Vendors may implement network and deployment topologies for distributing and replicating services in different clusters to overcome the availability and scalability limitations. The deployment topology assists in identifying the resource services mandatory for hardware and OS, since each service may require JDBC, reference libraries, etc. Multiple tenants retrieving the same hardware, application servers and databases may have different response and performance times leading to exhausted HTTP server, connectivity issues, dropped service requests, etc. To overcome this type of issue vendors, implement shared application servers based on tenant's transactions but this may increase the complexity with configuring, installing and automating the server deployments features.

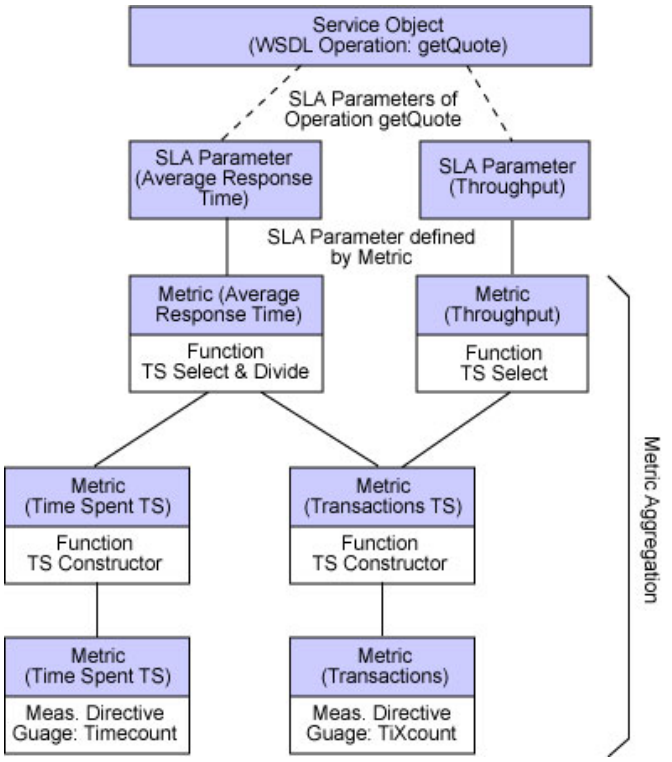

Figure 4. SLA parameter definition[17]

Each tenant applications and service are bound with an SLA, distributing a tenant's application on multiple clusters may increase the privacy, Identity and Access Management (IAM) and data breach-based risks as discussed above. As discussed in Section 2.2, 4.2 and 4.3 Multi-tenancy across different clusters is complicated, if the I4.0 tenant's services are processed at multiple regions, and services outsources to a third-party sub-contractor, this may only lead to monitoring, visibility, outages and control-based issues.

Figure 4 is adapted from IBM (for illustration purpose only) to explain the mechanism of SLA parameter definitions based on Web Service Description Language (WSDL). Considering a heterogeneous computing environment, where a single metrics assessment depends on multiple sub-metrics functionalities. With so many dependencies and complexity, the I4.0 cloud tenants will definitely end in a vendor lock-in. The WSDL toolkit may overcome SLA monitoring and privacy-based limitations in a tenant-vendor-subcontractor situation but will only be possible if three of them implement the same toolkit [25].

A cloud vendor may sub-contract tenant cloud processing to multiple third party sub-contractors based on the conditions of installing services on the sub-contractor's domain itself and may only make I4.0 tenants SLA visible which are being processed on the sub-contractor's domain. This model may hold the following possibilities: (i) Tenant, vendor, sub-contractor (ii) Multiple tenants, one vendor, one sub-contractor (iii) Multiple tenants, one vendor, multiple sub-contractors. Multiple sub-contractors may be added for same or different tenant cloud services by the cloud vendor subject to using the same web-service toolkit which is again a vendor lock-in situation.

4.4. Web Services

Services using standard XML messaging system and independent of operating systems and programming languages available over the internet are called web-services [15]. *"A web service is described as an application that accepts XML-formatted requests from other systems across the network"* [20]. Web service technology [16] *"depends upon specific XML standards such as: Simple Object Access Protocol (SOAP), XML-RPC (Request Procedure Call), or Representation State Transfer (REST) for messaging, Web Service Description Language (WSDL) for explaining the service interface, XML Schema for describing data types, and Universal Description Discovery and Integration (UDDI) for publishing and discovering service metadata"*. The two essential components for a web service are: self-describing/service publishing, self-discoverable/ser-vice identification and service execution [17].

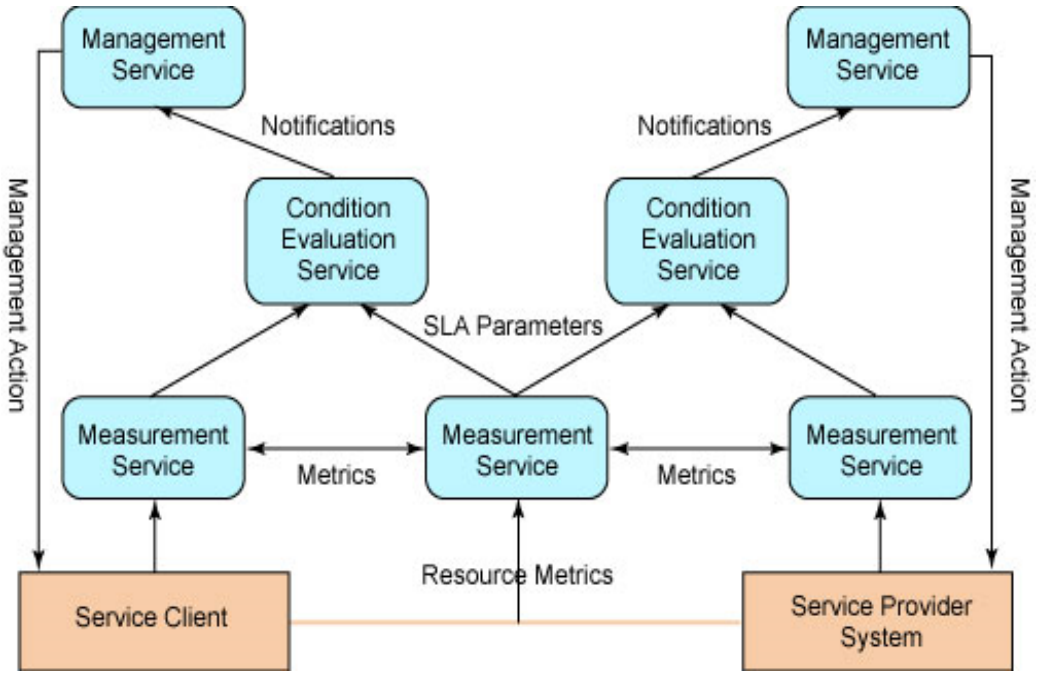

Figure 5. SLA monitoring model[17]

[20] www.guruteamirl.com/web-services

Web Service Description Language: The WSDL follows syntax-based web-services, missing auto discovery and composition features [18]. Since services cannot be automatically discovered, they cannot be invoked, this is where the semantic web services such as: Web Ontology Language for Services (OWL-S)[21] and Web-Service Modelling Ontology (WSMO) [19] may benefit as semantic web services can analyse key-words of similar meanings and invoking services. Figure 5 illustrates the model for SLA monitoring based on the WSDL toolkit. This model measures, evaluates and responds based on service conditions and violations, this service may also be extended to third-party subcontractor's (based on different services providing an abstracted SLA for gaining vendor control). Many examples of third party sub-contracting have been presented by [20] leading to vendor negligence on sub-contractor assessments. The sub-contractors were essentially hardware suppliers with no proper backup setups, loss of data because of inadequate quality storage and lack of expertise. Such risks may lead I4.0 tenants to major data security issues.

SLA service levels [21] are defined to assess performance levels of the deployed web-service based on a list of performance metrics such as: availability, latency, response times, etc. at the hardware, network, storage levels, etc.

4.5. Set of SLA management mechanisms addressing the SLA life-cycle

SLAs between a cloud tenant and vendor may function in stages of a life-cycle as illustrated in figure 6. Different vendors may follow the SLA life-cycle differently. For example, vendors may implement multiple services such as: deploy, provision and enforcing SLAs in a single stage or may interpret them with different names. The Web Service Agreement (WS-Agreement) and WSLA Language and Framework standards are widely applied for stating Service Level Agreements in an Service Oriented Architecture (SOA) [11]. The WSLA framework consists of SLAs in an Extensible Markup Language (XML) [22] file which is exchanged back and forth during the SLA lifecycle [12].

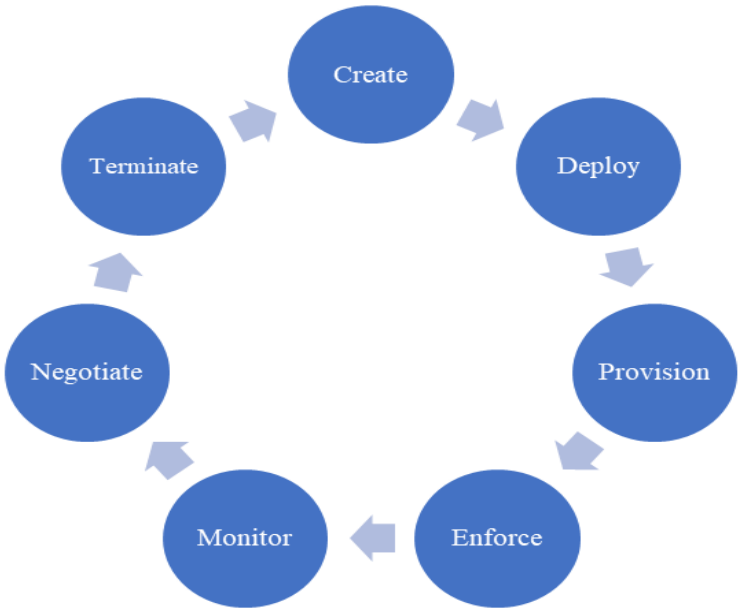

Figure 6. SLA Lifecycle

SLA Management Mechanisms: Vendors implement their pre-defined architectures to assist in monitoring compliance, provisioning, classifying the services and providing near real-time logs for multiple tenants, differentiating them based on the type of web service opted for[5]. This model however does not work in the hybrid multi-cloud ecosystems and hybrid third-party sub-contracting environments, since the tenants lack visibility and control.

Cloud vendors do not provide hybrid models therefore existing and possible issues related to SLA Orchestration and complexity have been discussed via author designed hybrid cloud models. The authors also felt that redesigning of application specific SLAs need to be addressed since every

[21] www.w3g.org/Submission/OWL-S

application is subject to different service levels, which ultimately affects the availability and other QoS parameters. The tenants deploying hybrid cloud realise the SLA QoS issues and limitations post-migration.

5. Cloud SLA Legal and Privacy aspect for Industry 4.0

The legal issues in the hybrid cloud environment are the most critical ones, few of which are as follows: (i) cloud vendors do not hold responsibility for any data loss, breach, malicious attack, unauthorized access, damage, etc. (ii) vendors processing tenant data in restricted geographical regions (iii) passing back and deleting the duplicated data after the vendor contract is over (iv) SLA exclusions (v) third-party sub-contracting liability (vi) credit terms, etc. Outsourcing data raises privacy concerns [23] due to lack of user control, visibility, transparency, multi-tenancy and VM based vulnerabilities, data governance risk and control, etc. Standards for Hybrid Clouds: Implementing cloud standards such as [12]: Open Cloud Computing Interface (OCCI), Open Grid Forum (OGF), Topology and Orchestration Standard for Cloud Applications (TOSCA), Cloud Data Management Interface (CDMI), etc. have become aggressive on tenant's demand and encouraging new tenants to adapt the cloud ecosystem.

Industry 4.0 is the next generation of smart factory involving new communication models and techniques. With thousands of IoT based devices transmitting data, insights, visibility, control and end-to-end security are a must requirement for SCADA (Industrial Control Systems). Industry 4.0 strictly adheres to the Industrial standards (i.e. IEC 62443 for cybersecurity, ISO 27001 for Information Security, etc.) but none of the cloud vendors follow a unified standardisation approach. Considering the above mentioned facts and SLA criteria, lack of knowledge/expertise in cloud SLAs may open up major security-based issues for Industry 4.0 tenants.

6. SLA Limitations in the Hybrid Cloud environment

Interoperable Service Level Agreements which may function independently with different cloud vendors and be understood irrespective of the platform limitations may be the only solution to overcome the existing issues highlighted in Section SLAs existing issues.

Currently, I4.0 cloud tenant migrations have only been possible at the application or database tier, since standardization and functionality has not been achieved at the web-service level yet. Cloud vendors individually deploying specific tools, implementation techniques and infrastructure governing software wish to maintain their monopoly and vendor lock-in situations, making the migrations for tenants from one platform to an-other difficult [24]. Some API designers intentionally build limitations to stop direct access to the internal features, limiting the scope of cloud vendors [25].

A wide number of projects [26] such as: Kubernetes, Docker, Mesos are working towards hybrid cloud standardization but this is yet formed in the container-oriented virtualization level. XML Schema (i.e. DProfSLA schema) for monitoring applications being under compliance with the promised SLA [27]. SLA Management mechanisms and architectures for hybrid cloud computing have been proposed by [28] [29] and [30] but these proposals have not been implemented by cloud vendors yet. Hybrid multi-clouds SLA complexity results in ambiguous operational management of the cloud, tenants find it hard to track and manage individual SLAs since the multiple applications running relate with each other. SLAs have multiple functionalities to provide: to adhere with tenant's application and data security, monitoring, visibility and control [22]. The core solution only lies in implementing a standard on the entire model, making it unified and easy to implement. As mentioned above, many vendors, projects and research has been proposed related to standardization but since it is an individual effort, the I4.0 tenants need to buy tools or implement frameworks or APIs to achieve the desired customization and flexibility in the hybrid cloud.

Extensible Stylesheet Language Transformations (XSLT) may be a possible solution for transforming different dialects of SLAs. XSLT[22] approach transforms XML data from one format to

[22] www.w3.org

another, the result form is an XML file, but it can change XML data into other formats by producing a XSLT stylesheet and processing the data. Changes required at the output can be achieved by making variations to the stylesheet and pro-cess it again. This provides benefits to nonprogrammers, who can alter the stylesheet and change the outcomes. Figure 7 illustrates a I4.0 cloud tenant, vendor and sub-contractor SLAs, since all parties are using different dialect of an XML, the XSLT based framework translates the SLAs into a unified one to understand different attributes, functionalities and specifications to be delivered. This framework acts as a mediator which translates information requested and responds back in the cloud actors (i.e. tenant, vendor, broker, auditor, etc.) own dialect.

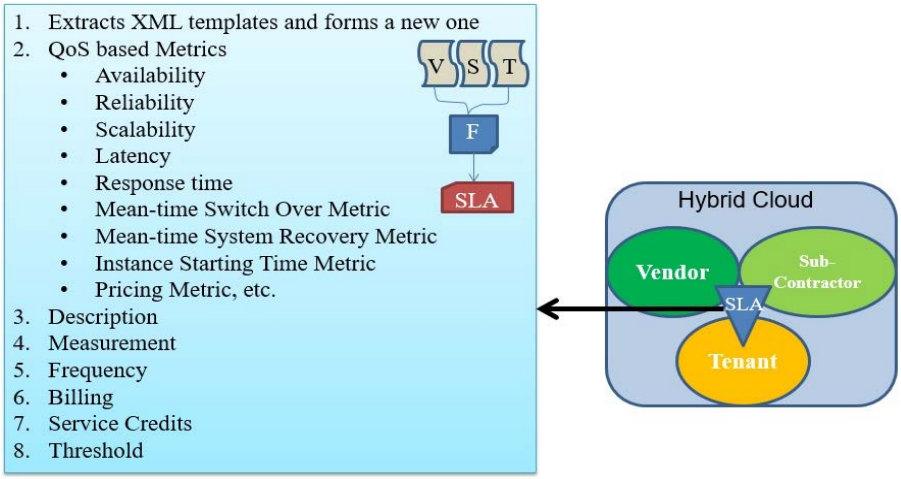

Figure 7. SLA Schema leading to framework designing

Few researchers [2][11][27] realized limitations associated to SLAs and designed frameworks for overcoming the existing issue. The frameworks were based on XML schemas have not been yet implemented due to vendor resistance and security-based issues. However, the WSLA framework is implemented by IBM WSDL Toolkit, since it provides interoperability only if the I4.0 tenant, cloud vendor and subcontractor use the same platform and not otherwise, thus leading to another vendor lock-in situation and leaving the issue unresolved.

This research was based on designing our own XML dialect, but was already done in past by some authors and has been limited to open source projects. The authors thus designed a Six Sigma Cloud Framework (SSCF) to overcome the cloud QoS issues. The Six Sigma Cloud Framework assists tenants/enterprises in being proactive, assessing the anticipated the Hybrid Cloud QoS before entering the cloud model. It also highlights the limitations, lock-ins and service breaches beforehand which helps the I4.0 tenants in negotiation and finding alternative solutions to overcome the hybrid cloud service breach. The fish bone cause-and-effect of the framework is presented by authors in [31]. Since this aspect of the project did not match with the dimension of this paper, the authors did not include this in the paper as it will form the basis of future publication.

Standardizing the Hybrid Cloud model would help in securing the system better. All vendors need to work towards a common goal to secure the Cloud IaaS making the deployment and I4.0 tenant adaption easier. Vendors resistance may only lead to alternate solutions of adapting and designing more cross-cloud platforms and applications, using transforming tools and frameworks.

7. Discussion

The study of uniform and standardised Service Level Agreements has been widely researched from different angles (i.e. toolkits, XML-based schemas, languages, standards, IT-operational management, etc.) since it can control and improve the cloud QoS. Cloud tenants have been actively looking for alternatives and solutions to mitigate vendor lock-ins, this is where the cross-cloud platforms tools, APIs and Frameworks assist, but cannot be considered as an ultimate solution since new cloud models are being commercialised without the underlying SLA being standardised.

8. Conclusion

Cloud Service Level Agreements play an vital role to manage and control the services leased by cloud vendors to the tenants. The Industry 4.0 environment depends on new communication technologies (IoT, Cloud Computing, AI, Robotics, etc). for automation, scalability, agility and process efficiency. In a self-driving, self-learning industrial environment, data security plays a vital role and needs to be secured by all ends (mediums over data is travelling and being processed). This is why it is important for cloud services to provide secure and standardised service. Cloud vendors conceal business and technical (underlying architectures, tools, management consoles, etc.) limitations, standards and QoS (availability, scalability, multi-tenancy and latency) issues smartly within the SLA and may go unnoticed by non-expert cloud tenants. Standardisation and transparency are compulsory for a successful cloud implementation. This paper acts as a catalyst bridging the SLA gaps between Industrial cloud tenants and vendors and developing insights on cloud QoS issues which otherwise stay disguised/unknown to the non-expert.

List of abbreviations

AWS	Amazon Web Services
CDMI	Cloud Data Management Interface
CSLA	Cloud Service Level Agreement
DGRC	Data Governance Risk and Control
DoS	Denial of Service
DSS	Decision Support System
DC	Data Center
EC2	Elastic Cloud Compute
EBS	Elastic Block Storage
ERP	Enterprise Resource Planning
GCP	Google Cloud Platform
GRS	Geographically Redundant Storage
HTTP	Hyper Text Transfer Protocol
IAAS	Infrastructure as a Service
IAM	Identity and Access Management
ITIL	IT Infrastructure Library
ISTM	Instance Starting Time Metric
ITSM	IT Service Management
JDBC	Java Database Connectivity
LRS	Locally Redundant Storage
OCCI	Open Cloud Computing Interface
OGF	Open Grid Forum
OLA	Operational Level Agreement
OS	Operating System
OWL-S	Web Ontology for Services
PAAS	Platform as a Service
QoS	Quality of Service
RA-GRS	Read-Access Geographically Redundant Storage
REST	Representation State Transfer
RPC	Request Procedure Call
SAAS	Software as a Service
SLA	Service Level Agreement
SLM	Service Level Management
SLO	Service Level Objectives
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
TB	Tera Byte

TOSCA	Topology and Orchestration Standard for Cloud Applications
URL	Uniform Resource Locator
VPN	Virtual Private Network
VM	Virtual Machine
WSDL	Web Service Description Language
WSMO	Web-Service Modelling Ontology
XML	EXtensible Markup Language
XSLT	EXtensible Stylesheet Language Transformations
ZRS	Zone Redundant Storage

Acknowledgement

The authors would like to acknowledge Science Foundation Ireland through the grant award (16/RC/3918) to the Confirm Centre for Smart Manufacturing and Johnson & Johnson, Automation Centre of Excellence (ACE) for part funding this research.

References

[1] Buyya, R, Broberg, J. and Goscinski, A. (2011). Cloud Computing Principles and Paradigms: John Wiley & Sons, Inc.

[2] Keller, A., Ludwig, H. (2002). Defining and Monitoring Service Level Agreements for dynamic e-Business. In: Proceedings of the 16th System Administration Conference (LISA 2002). Available: https://dl.acm.org/doi/10.5555/1050517.1050540.

[3] Dhirani, L. L., Newe, T., & Nizamani, S. (2020). Hybrid Multi-Cloud Demystifying SLAs for Smart City Enterprises Using IoT Applications. In IoT Architectures, Models, and Platforms for Smart City Applications (pp. 52-67). IGI Global. DOI: 10.4018/978-1-7998-1253-1.ch003.

[4] Paschke, A., Bichler, M. (2008) Knowledge representation concepts for automated SLA management. Decision Support Systems Vol. 46, pp. 187-205. Elsevier.

[5] Paschke, A., Kozlenkov, A., and Boley, H. (2007). A homogenous reaction rules language for complex event processing. International Workshop on Event Drive Architecture for Complex Event Process (EDA-PS 2007) Vienna, Austria.

[6] Erl, T., Mahmood, Z. and Puttini, R. (2013). Cloud Computing Concepts Technology & Architecture. Prentice Hall.

[7] Myerson, J. M. (2013). Best practices to develop SLAs for cloud computing. IBM Corporation. Available: https://www.ibm.com/developerworks/ cloud/library/cl-slastandards/.

[8] Morin, J. H., Aubert, J. and Gateau, B. (2012). Towards Cloud Computing SLA Risk Management: Issues and challenges. 45th Hawaii International Conference on System Sciences, Maui, USA, pp. 5509-5514, DOI: 10.1109/HICSS.2012.602.

[9] Dhirani, L. L., Newe, T., Lewis, E. and Nizamani, S. (2017). Cloud Computing and Internet of Things Fusion: Cost Issues. 11th International Conference on Sensing Technology (ICST 2017), Sydney, Australia, pp. 1-6, DOI: 10.1109/ICSensT.2017.8304426.

[10] Jin, H., Wang, X., Wu, S., Di, S. and Shi, X. (2015). Towards Optimized Fine-Grained Pricing of IaaS Cloud Platform. IEEE Transactions on Cloud Computing, vol. 3, no. 4, pp. 436-448, DOI: 10.1109/TCC.2014.2344680.

[11] Alhamad, M., Dillon, T. and Chang, E. (2010). Conceptual SLA Framework for Cloud Computing. 4th IEEE International Conference on Digital Ecosystems and Technologies, Dubai, United Arab Emirates, pp. 606-610, DOI: 10.1109/DEST.2010.5610586.

[12] Dhirani, L.L., Newe, T. and Nizamani, S. (2016). Tenant - Vendor and Third-Party Agreements for the Cloud: Considerations for Security Provision. International Journal of Software Engineering and Its Applications. vol. 10, no. 12, pp. 449-460, DOI: 10.14257/ijseia.2016.10.12.37

[13] Meiers, J. (2011). Best practices for cloud computing multi-tenancy Building a scalable, cloud computing multi-tenancy architecture. IBM Corporation, https://developer.ibm.com/depmodels/cloud/articles/cl-multitenantcloud/.

[14] Marinescu, D. C. (2013). Cloud Computing Theory and Practice. Elsevier.

[15] Cerami, E. (2002). Web Services Essentials. O Reilly Media Inc.

[16] Glass, G. (2002). Web Services Building Blocks for Distributed Systems. Prentice Hall.

[17] McIlraith, Sh., Son, T.C. and Zeng., H. (2001). Mobilizing the Semantic Web with DAML-Enabled Web Services. In Proceedings of the IJCAI 2001 Workshop on E-Business and the Intelligent Web, pp. 29–39, Seattle, WA.
[18] Le, D., Nguyen, V. and Goh, A. (2009). Matching WSDL and OWL-S Web services. IEEE Conference on Semantic Computing, Berkeley, USA, pp. 197-202, DOI: 10.1109/ICSC.2009.12.
[19] Lara, R., Roman, D., Polleres, A. and Fensel, D. (2004). A Conceptual comparison of WSMO and OWL-S. 2nd European Conference on Web Services, Lecture Notes in Computer Science, Springer, Berlin. Vol. 3250, pp. 254-269.
[20] Chow, R., Golle, P., Jackobsson, M., Shi, E., Staddon, J., Masouka, R., and Mollina, J. (2009). Controlling data on the cloud: outsourcing computations without outsourcing control. In Proc. Cloud Computing Security Workshop (CCSW09), pp. 85-90, DOI: 10.1145/1655008.1655020.
[21] Dan, A., Ludwig, H., Pacifici, G. (2003). Web service differentiation with service level agreements. IBM Corporation.
[22] Andrieux, A., Czajkowski, K., Dan, A., Keahey, K., Ludwig, H., Pruyne, J., Rofrano, J., Tuecke, S., and Xu, M. (2004) Web services agreement specification (WS-Agreement). In: Global Grid Form.
[23] Bernsmed, K., Jaatun, M. G., Meland, P. H., and Undheim, A. (2011). Security SLAs for Federated Cloud Services," 2011 Sixth International Conference on Availability, Reliability and Security, Vienna, 2011, pp. 202-209, DOI: 10.1109/ARES.2011.34.
[24] Sill, A. (2016). Standards for Hybrid Clouds. IEEE Cloud Computing, Vol 3, No. 1, pp 92-95.
[25] Sill, A. (2015). Socioeconomics of Cloud Standards. IEEE Cloud Computing, Vol 2, No. 3, pp. 8-11.
[26] Sill, A. (2015) When to Use-Standards-Based APIs (Part 2). IEEE Cloud Computing, Vol 2, No. 6, pp.80-84.
[27] Dusan, O., Milan, V., and Zora, K. (2012). Service Level Agreement XML Schema For Software Quality Assurance. Acta Technica Corviniensis - Bulletin of Engineering, Hunedoara, Vol. 5, No. 1, pp. 123-128 (2012).
[28] Patel, P., Ranabahu, A., and Sheth, A. (2009). Service Level Agreement in Cloud Computing. In Proceedings of OOPSLA. Available: https://corescholar.libraries.wright.edu/knoesis/78.
[29] Comuzzi, M., Kotsokalis, C., Rathfelder, C., Theilmann, W., Winkler, U., and Zacco, G. (2009) A framework for multi-level SLA management. In Proceedings of the ICSOC/ServiceWave 2009. Springer-Verlag, pp.187-196.
[30] Theilmann, W., Happe, J., Kotsokalis, C., Edmonds, A., Kearney, K., and Lambea, J. (2010). A Reference Architecture for Multi-Level SLA Management. Journal of Internet Engineering, Vol 4, No. 1,pp. 289-298. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.452.4283&rep=rep1&type=pdf.
[31] Dhirani, L. L., Newe, T., & Nizamani, S. (2018) Hybrid Cloud Computing QoS Glitches. In 2018 5th International Multi-Topic ICT Conference (IMTIC), pp. 1-6, DOI: 10.1109/IMTIC.2018.8467224.