
ABSTRACT

Cloud Computing has obvious extraordinary growth in the final few years. Although tremendous businesses see quite a few gaps like vendor lock-in with both Public Cloud choices as well as private Clouds leading to slower adoption than expected. Federated Clouds may be in a position to clear up the challenge. Cloud Federation brings together cloud resources from public as well as private environments so that companies can decide upon an environment on-demand that is smart for their specific workloads throughout carriers. While doing so, it additionally furnishes a unified access and examines of these assets enabling businesses to comfortably control, display and meters them. The intention of the paper is to fully grasp the necessity of a federated cloud and reference architecture to move about constructing one.

KEYWORDS: Cloud Computing, Cloud Federation, vendor lock-in.

INTRODUCTION

Cloud computing is a brand new paradigm for offering computing services on demand over the web. The vision that computing might be one days as five utility after (water, electricity, gas, and telephony) [3]. Computing services in Cloud may also be divided into, IaaS (Infrastructure-as-a-service), PaaS(Platform-as-a-provider) and SaaS (application-as-a-carrier). Persons in cloud computing eventualities can also be divided right into

- Cloud computing vendors,
- cloud computing consumers and
- End-users [14].

Cloud provider vendors own the bodily resources as datacentres. Cloud buyers; use these assets to provide carrier to users. And end-users access these services “anytime, at any place”. Computing offerings in cloud computing will be provide in keeping with the contract between Cloud service provider and consumer witch called SLA (provider degree contract). Conditions may arise the place a cloud provider is unable to furnish services agreed within the carrier degree agreement due to the dearth of resources or unexpected expand in incoming workload. In these instances, the cloud carrier provider rents the services from different cloud provider vendors and provides them to the customer. Hence the cloud service provider’s kind a union referred to as a federated cloud. Federation brings new industry opportunities for clouds .The place cloud vendors can buy and/or promote computing/storage capabilities and offerings to different Clouds

CLOUD SERVICES IN SHORT NOTE

Computing offerings in Cloud computing can be divided into three units:

- SaaS (application-as-a-provider): in this model, a whole utility is furnished to the purchaser, as a service on demand. Today SaaS is supplied by way of firms akin to Google, Salesforce, Microsoft, Zoho, and many others.
- PaaS (Platform-as-a-carrier): right here, a layer of application or development atmosphere is supplied as a carrier. The purchaser builds his possess applications using programming languages and instruments

supported via the supplier. Google's App Engine, force.Com, and so on are probably the trendiest PaaS examples.

- IaaS (Infrastructure-as-a-provider): client has full manipulated to manipulate underlying supplier's infrastructure corresponding to "Servers, storage systems, networking apparatus, information centre area and many others. Some usual examples are Amazon, GoGrid, three Tera, and so for

ECONOMIC THEORY

1. Vendor Lock- in
 2. Hold-up Problem
 3. Under_ investment
- Vendor Lock-in Supplier vendor lock-in is a drawback where a customer using a services or products cannot conveniently transition to a competitor's services or products. Seller lock-in is often the outcome of proprietary technologies which are incompatible with those of competitors. Nonetheless, it may also be precipitated by inefficient techniques or contract constraints, among other matters. The fear of vendor lock-in is normally noted as a most important impediment to cloud service adoption. The complexities of cloud provider migration imply that many shoppers stick with a supplier that doesn't meet their desires, simply to avoid the cumbersome system. To maneuverer knowledge from one provider's cloud atmosphere to an extra, for example, it's most likely necessary to first transfer the information back to the consumer's web site after which move it to the new provider's atmosphere. Furthermore, the information could have been altered for compatibility with the usual supplier's system in order that what's again to the purchaser wishes to be returned to its former state before it may be moved again. The excellent option to preclude supplier lock-in is to opt for your service accurately in the first place, following tips for keeping off cloud vendor lock-in: Learn the satisfactory print of each and every provider's policies, and if quintessential, ask them instantly how they facilitate moving purchaser data out of their cloud storage repository. Ask the supplier whether or not they present information migration instruments or offerings to facilitate the motion of colossal quantities of data. Pick providers that have pledged to aid emerging industry specifications, such as the Cloud data management Interface (CDMI) commonplace created with the aid of the Storage Networking industry association (SNIA).
 - Hold-up Problem bargaining vigor has shifted to the provider .He can negotiate higher charges.
 - Under Investment Risk of Lock-in Anticipation that Bargaining energy will shift A get together being forced to be given sub-gold standard conditions are not able to break out the obstacle as a result of the lock-in and finds itself in a keep-up. Illustration: Closed PaaS

MOTIVATION FOR CLOUD FEDERATION

- Avoidance of dealer lock-in: By means of making use of a couple of clouds and being in a position to freely transit Workload amongst them, a cloud client can easily avert vendor lock-in. In case a provider changes coverage or pricing that have an impact on negatively its clients, client could without problems migrate in other places
- Multitude of players: That is the critical piece to the very definition of federated cloud ecosystems. This may be certain that the market vulnerability, monopoly, doesn't develop within the process. One of the most imperative conditions for making certain this kind of participation via provider providers is to have at least one open source cloud platform.
- Heterogeneity of cloud systems: An extra major characteristic is the presence of heterogeneous platforms in the ecosystem. A federated cloud ecosystem can also be realized simply with one cloud platform so long as the platform supports federation however it'll just shift the monopoly from the provider providers to the platform. Monopoly at any layer is dangerous for the users and this characteristic ensures that we prevent monopoly on the platform layer. I don't help monopoly even though the underlying platform is open supply. The proposal of heterogeneity not only helps prevent any monopoly, it is relevant to handle the needs of lengthy tail users. Extra importantly, as i will provide an explanation for in a future post or research report, it's going to help cloud consumers hinder the so called cascading screw ups (Cloud is an elaborate adaptive procedure and heterogeneity is crucial to preclude cascading screw ups in such systems) in IT.

- Interoperability and Portability: For a federated system to satisfy the definition of cloud computing and to restrict lock-in, the cloud systems need to be interoperable. Also, users must be equipped to maneuverer their workloads and knowledge between specific providers. I don't care if it works by means of standardization of APIs and formats or via 0.33 party services/libraries however it is a critical characteristic.
- No supplier lock-in: The consequences of the above three conditions will make sure that there's no supplier lock-in. However, I want to state this as separate attribute behaviour of federated cloud ecosystems.
- Geographical distribution: That is additionally a major characteristic of federated cloud ecosystems. Even the requirement of multiple providers in the market is not ample to satisfy the world's computing needs, certainly those that fall within the lengthy tail neighbourhood. Also, the regulatory regimes aren't going away every time quickly (absolutely, not in our lifetime). As a way to meet these desires/specifications, there is also a necessity for geographical distribution of providers/datacentres worldwide.

DESIGNING OF FEDERATED CLOUD

Designing Federated Cloud offerings from distinctive vendors aggregated in a single pool supporting three basic interoperability elements:

1. Resource migration
2. Resource redundancy
3. Combination of complementary assets & services

- **Redundancy vs Migration**

Table 1: Comparison of Redundancy and Migration

Redundancy	Migration
Is used each time there's a subset of (accurately organized) carrier choices that provide better utility to a client than any single carrier supplying	can be brought about when a new service supplying offers better utility to a patron than any previously used service delivering

- **Resource Redundancy**
- **IaaS**
- Compute Resources
- Redundant Deployment
- Redundant Computation
- Parallel Computation
- Storage Resources
- Replication
- Fragmentation
- Redundant Deployment
- Application logic is deployed to distinct vendors.
- Incoming requests are processed by way of just one example.
- Increase the supply whilst lowering supplier dependence.
- Compliance with rules, which require situations.
- client proximity would be an hassle to decrease latency

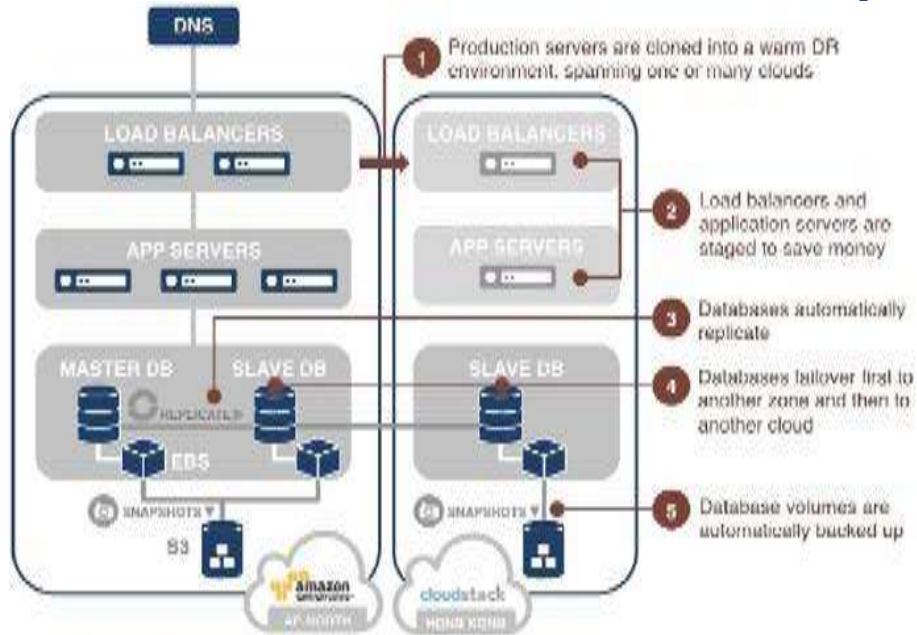


Figure 1: Resource Redundancy: Redundent Deployment Architecture [21]

- Redundant Computation
 - application common sense is deployed to extraordinary providers
 - each request is processed with the aid of a couple of illustration
 - enhance performance by way of reducing the chance of an example failing proper earlier than completing an assignment
- Parallel Computation
 - The info is broken down at bit level and processed at distinct vendors' sites following the identical software logic or complimentary services are deployed to one-of-a-kind providers.

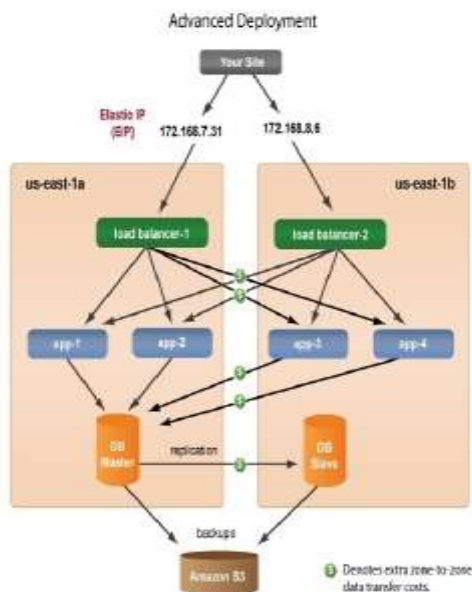


Figure 2: Resource Redundancy: Redundant Computations Architecture [21]

PaaS

- hard to use redundantly
- As they most commonly comply with another programming mannequin
- Aid simplest a constrained quantity of programming languages
- Limitations on the programming mannequin they build upon in order that applications need to be satisfactory-tuned for a exact platform.
- Illustration: Google App Engine, drive.Com
- Open source PaaS
- Hosted by way of the customer on high of IaaS
- illustration: VMWare Cloud Foundry, Redhat Open Shift

SaaS

- Very difficult and more commonly fee-intensive to federate intricate offerings like Salesforce.
- Pretty much inconceivable to isolate smaller constructing blocks of the provider as no competing solutions exist, which present exactly the equal performance.
- The possibly proprietary data codecs and APIs of such services expand the challenge

- Resource Migration

Migration incorporates scenarios where resources are being transferred from one Cloud provider A to another Cloud provider B.

- Shadowed or redundant migration
- Non-redundant migration
- Full Migration
- Partial migration

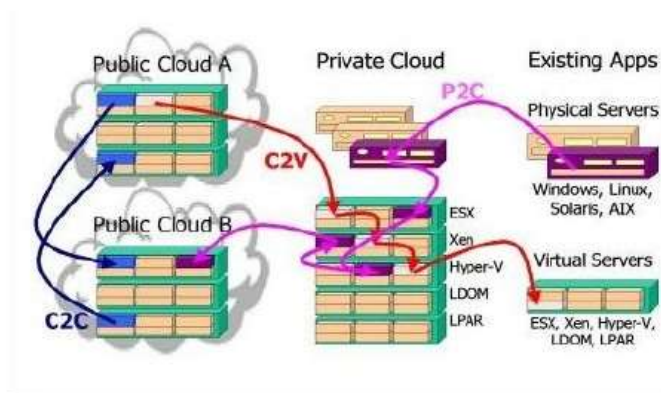


Figure 3: Resource Migration Architecture [21]

- **Reference Architecture**

- Interoperable systems
 - suitable Cloud management techniques
Example: Eucalyptus, Amazon EC2, Cloudstack Cloudbridge
 - Multi-Cloud Libraries
Example: Libcloud – Python, jCloud – Java, Fog – Ruby
- Provisioning Engine
 - Discover utility packages and components.

- Map trade accessories to pool of assets.
- Generate Deployment descriptors established on user requisites and present learned resources and policies
- Resource Manager
 - control the entire assets in a unified method
 - provide an abstraction of the APIs of the underlying services
 - enable configurations of assets in unified means making use of adaptors like multi-cloud libraries
- Deployment supervisor
 - Founded on a deployment description the supervisor executes useful resource administration commands via resource Managers.
 - It ensures the supply and the correct configuration of provisioned resources.
 - Performs transformation of pictures throughout virtualized environments.
 - provide ensures and SLAs
- Configuration manager
 - Recreate digital home equipment and software applications situated on current configurations.
- Information Distribution manager
 - Setup information storage as per configuration - information replication, information redundancy, information Sharing
- Monitoring techniques
 - collect expertise about resource states
 - document sudden conditions

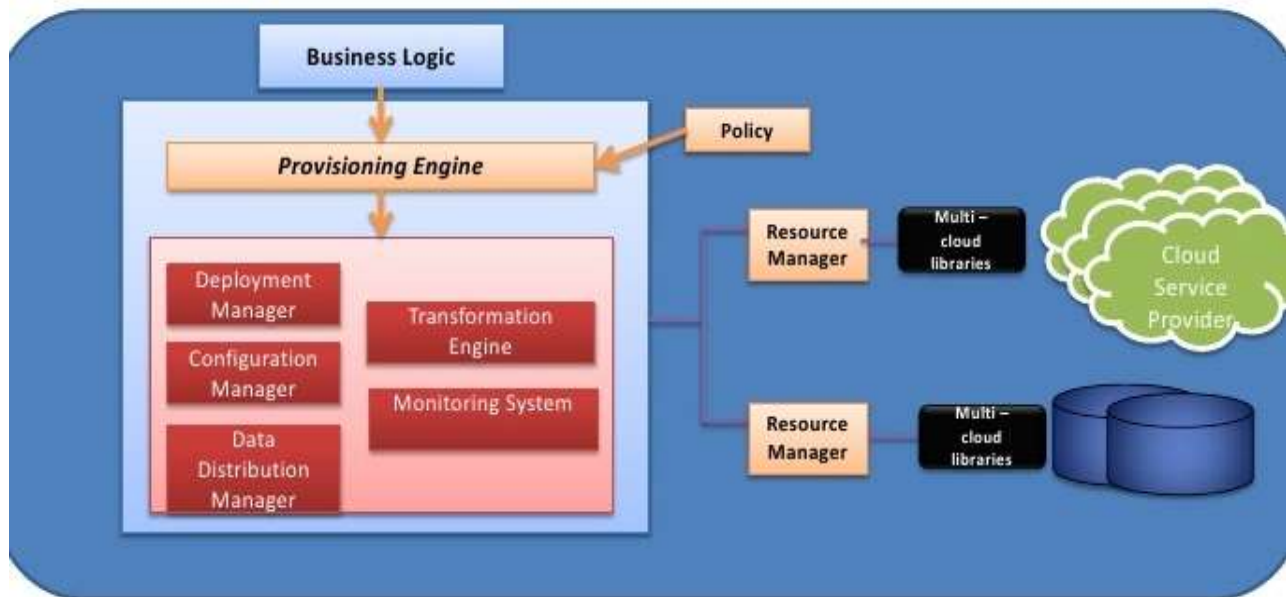


Figure 4: Reference Architecture [21]

The intention of the paper is to fully grasp the necessity of Federated cloud and reference architecture to move about constructing one by understanding the concepts of Economic Theory. Apart from this it helps Enterprises to bring resources and access them .Future work is about configuring or constructing group data sharing in Federated clouds.

REFERENCES

- [1] Bernstein, Deepak Vij, and Stephen Diamond. 2011. An intercloud cloud computing economy - technology, governance, and market blueprints. In Proceedings of 2011 Annual SRII Global Conference (SRII). IEEE, San Jose, CA, 293–299
- [2] Buyya, R., Ranjan, R., & Calheiros, R. N. (2010). Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services. In Algorithms and architectures for parallel processing (pp. 13-31). Springer Berlin Heidelberg.
- [3] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation computer systems, 25(6), 599-616
- [4] Carlini E, Coppola M, Dazzi P, Ricci L, Righetti G. Cloud Federations in Contrail. Proceedings of Euro-Par 2011: Parallel Processing Workshops, vol. 7155, Alexander Mea (ed.). Springer Berlin / Heidelberg: Bordeaux, France, 2012; 159–168
- [5] Celesti, A., Tusa, F., Villari, M., & Puliafito, A. (2010, July). How to enhance cloud architectures to enable cross-federation. In Cloud Computing. (CLOUD), 2010 IEEE 3rd International Conference on (pp. 337-345). IEEE
- [6] Distefano, S., Cunsolo, V. D., Puliafito, A., & Scarpa, M. (2010). Cloud@ home: A new enhanced computing paradigm. In Handbook of Cloud Computing (pp. 575-594). Springer US
- [7] Ferrer, A. J., Hernández, F., Tordsson, J., Elmroth, E., Ali-Eldin, A., Zsigri, C., & Sheridan, C. (2012). OPTIMIS: A holistic approach to cloud service provisioning. Future Generation Computer Systems, 28(1), 66-77.
- [8] Goiri, I., Guitart, J., & Torres, J. (2010, July). Characterizing cloud federation for enhancing providers' profit. In Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on (pp. 123-130). IEEE
- [9] Global Inter-Cloud Technology Forum. Use Cases and Functional Requirements for Inter-Cloud Computing. Technical Report, Global Inter-Cloud Technology Forum 2010
- [10] Grozev, N., & Buyya, R. (2012). InterCloud architectures and application brokering: taxonomy and survey. Software: Practice and Experience
- [11] Tobias Kurze, Markus Klemsy, David Bermbachy, Alexander Lenkz, Stefan Taiy, and Marcel Kunze. 2011. Cloud federation. In Proceedings of the 2nd International Conference on Cloud Computing GRIDS, and Virtualization. 32–38.
- [12] Mahmood, Z. (2011, August). Cloud computing: Characteristics and deployment approaches. In Computer and Information Technology (CIT), 2011 IEEE 11th International Conference on (pp. 121-126). IEEE.
- [13] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing (draft). NIST special publication, 800(145), 7
- [14] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. 2010. A view of cloud computing .Common. ACM53, 4 (2010), 50–58
- [15] Ranjan, R., & Buyya, R. (2008). Decentralized overlay for federation of enterprise clouds. arXiv preprint arXiv:0811.2563
- [16] Ranjan, R., Buyya, R., & Nepal, S. (2013). Model-driven provisioning of application services in hybrid computing environments. Future Generation Computer Systems, 29(5), 1211-1215.
- [17] Rochwerger, B., Breitgand, D., Levy, E., Galis, A., Nagin, K., Llorente, I. M., ... & Galan, F. (2009). The reservoir model and architecture for open federated cloud computing. IBM Journal of Research and Development, 53(4), 4-1
- [18] Rochwerger, Constantino V´ azquez, David Breitgand, David Hadas, Massimo Villari, Philippe Massonet, Eliezer Levy, Alex Galis, Ignacio M. Llorente, Rub´ en S. Montero, Yaron Wolfsthal, Kenneth Nagin, Lars
- [19] Larsson, and Ferm´ in Gal´ an. 2011. An architecture for federated cloud computing. In Cloud Computing.

[20] John Wiley & Sons, 391–411.

[21] Calcavecchia, N., Celesti, A., & Nitto, E. (2012). Understanding Decentralized and Dynamic Brokerage in Federated Cloud Environments. In M. Villari, I. Brandic, & F. Tusa (Eds.), *Achieving Federated and Self-Manageable Cloud Infrastructures: Theory and Practice* (pp. 36-56)

[22] David Bernstein and Deepak Vij. 2010b. Intercloud Security Considerations. In *Proceedings of the Second IEEE*

[23] International Conference on Cloud Computing Technology and Science (CloudCom '10). Indianapolis, USA, 537–544

[24] Source from www.clogeny.com