ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 1, January 2013

A Survey on Recent Trends in Cloud Computing and its Application for Multimedia

M. Ramya Catherine, E.Bijolin Edwin

Abstract--- **Cloud computing has been the emerging technology in the recent years and computing has shifted it base to the clouds taking the world of computing to cloud computing. This paper surveys some of the recent technologies used in the cloud computing. We present the methodologies used and the advantages and disadvantages of the various methods proposed.**

Index Terms--- Cloud computing, Cloud Media Streaming, Flexible edge cloud, Media Cloud

I. INTRODUCTION:

Cloud computing is the use of computing resources, hardware as well as software, that are delivered as a service through a network, typically the Internet. Cloud computing is a universal collection of data which extends over the internet in the form of resources and forms individual units within the environment of virtualization. Held together by providers of infrastructure, service providers and also the consumers, it is then semantically accessed by various users. Through virtualization technology, it is possible to run just not an application, but also a full server within the Cloud, which reduces the cost of the hardware base in a typical one service, that is, one server model.

Cloud computing possesses the following key characteristics:

Agility improves with users' ability of re-provisioning technological infrastructure resources.

Application programming interface *(API)* provides accessibility to software that enables machines to interact with cloud software just as the user interface facilitates interaction between humans and computers. Cloud computing systems generally use REST-based APIs.

Cost is claimed to be reduced and low, and in a public cloud delivery model, capital expenditure is changed to operational expenditure. This is purported to lower barriers to enter as infrastructure is usually provided by a third-party and does not need to be purchased for one-time or infrequent intensive computing tasks. Pricings on a utility computing basis are fine-grained with usage-based options and fewer IT skills are required for implementation. The e-FISCAL projects' state of art repository contains several articles focusing on the cost aspect in detail, most of them concluding that cost savings depend on the type of activities supported and infrastructure available in house.

Device and location independence enable users to access systems through a web browser irrespective of their location or what device they are using. As infrastructure is off-site (typically provided by a third-party) and accessed through the Internet, users can establish connection from anywhere.

Virtualization technology allows servers and storage devices to be shared and utilization to be increased. The applications can be easily migrated from one physical server to any other physical server.

Multi-tenancy provides sharing of resources and cost across a large pool of users and thereby allowing:

Infrastructure Centralization in locations with lower costs (like real estate and electricity).

Peak-load capacity to increase (users need not engineer for highest possible load-levels)

Utilization and efficiency improvisations for systems that are often only 10 to 20% utilized.

Reliability is improved if multiple redundant sites are used, that make well-designed cloud computing suitable for business continuity and disaster recovery.

Scalability and elasticity via dynamic ("on-demand") provisioning of resources on a fine-grained, self-service basis, close to real-time, without users engineering for peak loads.

Performance is monitored, is consistent and the loosely coupled architectures are constructed using web services as the system interface.

Security could improve due to centralization of data and increased security-focused resources but concerns can persist about loss of control over certain sensitive data, and unsecured stored kernels. Security is often better than other traditional systems, partially because providers are able to devote resources to solve security issues that many customers cannot afford. However, the security complexity is greatly increased when data is distributed over a greater number of devices and in multi-tenant systems that are being shared by unrelated users. Moreover, users' access to security audit logs might be difficult or impossible. Private cloud installations are partially motivated by users' desire to retain control over the infrastructure and to avoid losing control of information security.

M. Ramya Catherine, *Department of Information Technology, Karunya University, Post Graduate Scholar Coimbatore, Tamil Nadu, India. +91 99443 77754*

E. Bijolin Edwin, *Department of Information Technology, Karunya University, Assistant Professor Coimbatore, Tamil Nadu, India. +91 94861 29746*

Maintenance of applications in cloud computing is easier, because they do not need separate installation on each user's computer and also, they can be accessed from different places.

Cloud providers offer cloud services under three fundamental models: Infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS) where IaaS is the most basic and each higher model abstracts from the details of the lower models.

Following are the types of public cloud computing:

- Infrastructure as a service (IaaS),
- Platform as a service (PaaS),
- Software as a service (SaaS)
- Storage as a service (STaaS)
- Security as a service (SECaaS)
- Data as a service (DaaS)
- Business process as a service (BPaaS)
- Test environment as a service (TEaaS)
- Desktop as a service (DaaS)
- API as a service (APIaaS)

Out of the above, three are the basic services. They are discussed below.

A *Platform as a Service (PaaS)*

Cloud computing has evolved to include platforms for building and running custom web-based applications, known as Platform-as-a- Service. PaaS is an outgrowth of the SaaS application delivery model. The PaaS model makes all facilities required to support the complete life cycle of building and delivering web applications and services entirely available from the Internet, all without any software downloads or installation for developers, IT managers, or end users. Unlike the IaaS model, where developers may create a specific operating system instance with homegrown applications running, PaaS developers are concerned only with web based development and generally do not care what operating system is used. PaaS services allow users to focus on innovation rather than complex infrastructure. Organizations can redirect a significant portion of their budgets to creating applications that provide real business value instead of worrying about all the infrastructure issues in a roll-your-own delivery model. The PaaS model is thus driving a new era of mass innovation. Now, developers around the world can access unlimited computing power. Anyone through Internet can build powerful applications and easily deploy them to users globally. PaaS offers a faster, more cost-effective model for application development and delivery. PaaS provides with the entire infrastructure needed to run applications over the Internet. Such is the case with companies such as Amazon.com, eBay, Google, iTunes, and YouTube. The new cloud model has made it possible to deliver such new capabilities to new markets via the web browsers. PaaS is based on a metering or subscription model, so users pay only for what they use. PaaS offerings include workflow facilities for design and development, testing, deployment, and hosting of applications, as well as application services like team collaboration, database integration, scalability, security, storage, persistence, virtual offices, etc.

Key Characteristics of PaaS

Chief characteristics of PaaS include services to develop, test, deploy, host, and manage applications to support the application development life cycle. Web-based user interface creation tools typically provide some level of support to simplify the creation of user interfaces, based either on common standards such as HTML and JavaScript or on other, proprietary technologies. Supporting a multitenant architecture helps to remove developer concerns regarding the use of the application by many concurrent users. PaaS providers often include services for concurrency management, scalability, fail-over and security. Another characteristic is the integration with web services and databases. Support for Simple Object Access Protocol (SOAP) and other interfaces allows PaaS offerings to create combinations of web services (called mashups) as well as having the ability to access databases and reuse services maintained inside private networks. The ability of forming and sharing code with ad-hoc, predefined, or distributed teams greatly enhance the productivity of PaaS offerings. Integrated PaaS offerings provide an opportunity for developers to have much greater insight into the inner workings of their applications and the behavior of their users by implementing dashboard like tools to view the inner workings based on measurements such as performance, number of concurrent accesses, etc. Some PaaS offerings leverage this instrumentation to enable pay-per-use billing models.

B *Software as a Service (SaaS)*

The traditional model of software distribution, where software is purchased for and installed on personal computers, is occasionally referred to as Software-as-a-Product. Software-as-a-Service is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, generally the Internet. SaaS is an increasingly prevalent delivery model as underlying technologies that support web services and service-oriented architecture (SOA) have matured and new developmental approaches became popular. SaaS is also a pay-as-you-go subscription licensing model. Meanwhile, broadband services have become increasingly available to support user access for larger cover around the world. The huge strides made by Internet Service Providers (ISPs) to increase bandwidth, and the constant introduction of ever more powerful microprocessors coupled with inexpensive data storage devices, is providing a huge platform for designing, deploying, and using software across all areas of business and personal computing. SaaS applications must be able to interact with other data and other applications in an equally wide variety of environments and platforms. SaaS is closely related to other service delivery models we have described. IDC identifies two slightly different delivery models for SaaS. The hosted application management model is similar to an Application Service Provider (ASP) model. Here, an ASP hosts commercially available software for customers and delivers it over the Internet. The other model is a software on demand model where the provider gives customers network-based access to a single copy of an application created specifically for SaaS distribution. SaaS is most often implemented to provide business software functionality to enterprise customers at a low cost while allowing those customers to obtain the same benefits of software that is commercially licensed, internally operated without the associated complexity of installation, licensing, support, management and high initial cost. Most customers have little interest in the how or why of software implementation, deployment, etc., but all have a need to use software in their work. Many types of software are well suited to the SaaS model (e.g., accounting, customer relationship management, email software, human resources, IT security, IT service management, video conferencing, web analytics, web content management). The distinction between SaaS and earlier applications delivered over the Internet is that SaaS solutions were developed specifically to work within a web browser. The architecture of SaaS-based applications is specifically designed to support many concurrent users (multitenancy) at once. This is a big difference from the traditional client/server or application service provider (ASP) based solutions that cater to a contained audience. SaaS providers, on the other hand, leverage enormous economies of scale in the deployment, support, management and maintenance of their offerings.

Key Characteristics of SaaS

Deploying applications in a service-oriented architecture is a more complex problem than is usually encountered in traditional models of software deployment. As a result, SaaS applications are generally priced based on the number of users that can have access to the service. There are often additional fees for the use of help desk services, extra bandwidth, and storage. SaaS revenue streams to the vendor are usually lower initially than traditional software license fees. However, the trade-off for lower license fees is a monthly recurring revenue stream, which is viewed by most corporate CFOs as a more predictable gauge of how the business is faring quarter to quarter. These monthly recurring charges are viewed much like maintenance fees for licensed software. The key characteristics of SaaS software are the following:

Network based management and accessing commercially available software from central locations rather than at each customer's site, enabling customers to access applications remotely via the Internet.

Delivery of application from a one-to-many model (single-instance, multitenant architecture), as opposed to a traditional one-to-one model.

Centralized enhancement and patch updating that obviates any need for downloading and installing by a user. SaaS is often used in conjunction with a larger network of communications and collaboration software, sometimes as a plug-in to a PaaS architecture.

C. Infrastructure-as-a-Service (IaaS)

*I*aaS is the delivery of computer infrastructure (typically a platform virtualization environment) as a service. IaaS leverages significant technology, data center and services investments to deliver IT as a service to customers. Unlike traditional outsourcing, that requires extensive due diligence, negotiations ad infinitum, and complex, lengthy contract vehicles, IaaS is centered around a model of service delivery that provisions a predefined as well as standardized infrastructure specifically optimized for the customer's applications. Rather than purchasing data center space, network equipment, servers, software, etc., IaaS customers rent those resources as a fully outsourced service. Usually, the service is billed on a monthly basis, just like a utility company bills customers. The customer needs to pay only for resources consumed. The chief benefits of using this type of outsourced service include:

Use of the latest technology for infrastructure equipment

Secured, "sand-boxed" (protected and insulated) computing platforms that are usually security monitored for breaches.

Reduced risk by having off-site resources maintained by third parties

Ability to manage service-demand peaks and valleys

Lower costs that allow expensing service costs instead of making capital investments

II CREATING A 'CLOUD STORAGE' MASHUP FOR HIGH PERFORMANCE, LOW COST CONTENT DELIVERY [1]

Many 'Cloud Storage' providers have launched in the last two years, providing data storage which can be accessed through internet and delivered in several continents that is backed by rigorous Service Level Agreements (SLAs), guaranteeing performance and uptime targets. The facilities that these providers offer are leveraged by developers via provider-specific Web Service APIs. For content-creators, these providers are a genuine alternative to dedicated Content Delivery Networks (CDNs) for global file storage and delivery, as they are much cheaper, have considerable performance and no ongoing contract obligations. As a result, the idea of utilizing Storage Clouds as a 'poor mans' CDN is very enticing. However, many of these 'Cloud Storage' providers provide just the basic storage services, and offer not the capabilities of a fully-featured CDN such as intelligent replication, load redirection, failover and load balancing. Moreover, they may be difficult to use for non-developers, as each service is best utilized via unique web services or programmer APIs. This paper describes the design, architecture, implementation and user-experience of Meta Content Delivery Networks, a system that integrates these 'Cloud Storage' providers into a unified Content Delivery Network service that provides high performance, low cost, content storage which is geographically distributed and delivery for content creators, and is also managed by an easy-to-use web portal.

Advantages:

- Intelligent deployment of content
- Viewing, modifying or deleting existing content deployment.
- Maximized coverage and performance
- Cost and QoS optimized deployment

Disadvantages:

- No user customization
- It does not enhance the user's experience of the internet.

III. DESKTOP TO CLOUD TRANSFORMATION PLANNING [2]

Traditional desktop delivery model is based on a large number of distributed PCs executing operating system and desktop applications. Management of traditional desktop environments is indeed a challenge and costly. Tasks such as installations, changes in configuration, security issues require procedures that are time-consuming and dedicated desk-side support. And moreover, these distributed desktops are usually underutilized, resulting in low ROI. Moreover, this distributed desktop computing model also creates a security concern as sensitive information could be compromised with stolen laptops or PCs. Desktop virtualization, moves computation to the data center and allows users to access their applications and data using stateless .thin-client devices and therefore alleviates some of the problems of traditional desktop computing. Enterprises can thus leverage the flexibility and benefits by costs of running users' desktops on virtual machines hosted at the data center to enhance business agility and reduce business risks while lowering TCO. Recent research and development in cloud computing provides openings in mass hosting of desktops and providing them as a service. However, transforming legacy systems to desktop clouds and provisioning proper capacity is a challenging problem. Desktop cloud must be appropriately designed and provisioned to offer low response time and good working experience to desktop users while optimizing back-end resource usage and therefore minimizing provider's costs. This paper illustrates the tools and approaches developed to facilitate fast and accurate planning for desktop clouds.

Advantages:

- Desktop virtualization is done
- Desktop benchmarking is the primary concept and is done
- User profiling helps the cloud to distinguish between the resources of interests.

Disadvantages:

- There's no user customization
- Neither there is enhanced use of the internet in the user's experience.

IV TOWARDS SERVICE COMPOSITION BASED ON MASHUP: [3]

Mashup is a hallmark of Web 2.0 and attracts both industry and academia recently. It refers to an ad hoc composition technology of Web applications that allows users to draw upon content retrieved from external data sources to create entirely new services. In comparison to traditional "developer-centric" composition technologies, e.g., BPEL and WSCI, mashup provides a flexible and easy-of-use way for composition of services on web. It makes the consumers free to compose services as they wish as well as simplifies the composition task. This leads to two contributions. Firstly, the mashup architecture, extends the current SOA model with mashup and analyzes how it facilitates the service composition. Secondly, a mashup component model to help developers leverage to create their own composite services.

Advantages:

- *Mo*re reusable
- Web based
- Light weight
- End consumer centric

Disadvantages:

- Data pollution issues
- Integration issues when screen scrapping techniques must be used for data acquisition.

V. A CLOUD MULTIMEDIA PLATFORM: [4]

Social networking web applications such as Facebook and Flickr present new challenges for storing and processing user generated content, i.e. multimedia. Handling massive amounts of data requires special systems that need upfront investment, which may hinder the realization of new innovative ideas. Instead, cloud computing as a new emerging operations model promises to deliver elastic on-demand unlimited computing resources as a utility. This paper proposes architecture for a Cloud Multimedia Platform that does the heavy-lifting for massive amounts of multimedia storage and processing in the spirit of the cloud computing paradigm.

Challenges:

- Scalability
- Elasticity
- Abstraction
- Simplicity
- Interoperability
- Distributed data management
- Responsive services
- Service orchestration

VI NETWORK EDGE INTELLIGENCE FOR THE EMERGING NEXT-GENERATION INTERNET: [5]

The success of the Content Delivery Networks (CDN) in the recent years has demonstrated the increased benefits of the deployment of some form of "intelligence" inside the network. Whereas, cloud computing, has the benefits of scale economies and the use of a generic infrastructure to support a variety of services. Following that, this paper proposes to move away from the smart terminal-dumb network dichotomy to a model where some degree of intelligence is put back within the network, specifically at the edge, with the help of Cloud technology. This paper proposes the deployment of an Edge Cloud, integrating a variety of user-side as well as server-side technologies. On the user side, surrogate which is an application, running on top of the Cloud, supports the virtual client. The underlying network infrastructure is hidden by the surrogate layer from the user, thereby allowing simpler and easily managed terminals. Network side services which support delivery and exploit content are also deployed on this infrastructure, thereby giving the Internet Service Providers (ISP), opportunities to become directly involved in content and service delivery.

Challenges:

- Complexity
- Security
- Heterogeneity
- Mobility and service continuity
- Integration of the service with support for content delivery in a unified infrastructure.

Advantages:

- User-side virtualization reduces the complexity of managing the computers.
- Gives uniform access from different locations and multiple devices.
- Simple user exchange
- Enhanced security and billing
- Enhanced user experience
- Reuse of mashup content

Disadvantages:

- Issues regarding complexity, compatibility and integration are on the client side.

VII GIVING USERS AN EDGE: A FLEXIBLE CLOUD MODEL AND ITS APPLICATION FOR MULTIMEDIA:[6]

The objective of the paper is to enhance the user's experience of the Internet by moving the desktop functionality within the network in the Void environment. This is achieved by the introduction of a flexible edge cloud which allows to the users expand the benefits of using the Internet cloud model. The main focus is on media processing and user customization.

This paper presents the Edge Cloud architecture and explores how the user's experience of the Internet is enhanced, by moving desktop functionality into the network, in the Void environment. It explores how this model allows us to expand the range of benefits of using the Internet Cloud model, especially when media processing and user customization are concerned. The implementation of a prototype, which transcodes audio and video stream inside the Cloud is presented. The prototype has been deployed in two different environments: in a local lab and in the Amazon EC2 public Cloud. The performance of the prototype by analyzing the inter-arrival jitter is also studied.

A. *Edge Cloud:*

The Edge Cloud introduces the surrogate layer, on top of the traditional (storage and computing) core Cloud services. The surrogate layer acts as a relay for the user to access, organize, provide and monitor different composite Cloud-based services, of diverse nature.

The Edge Cloud, with this surrogate layer, helps us move away from the enterprise-based, mostly data intensive use of the Cloud to meet the wider needs of the general public.

B. *Cloud-based user-oriented applications (Void)*

Void, an Edge Cloud-based framework supports user-controlled coordination of Cloud-based applications and its use in media processing. Void's purpose is to support the Cloud-based virtualization of user applications typically running on the desktop, which require exchange of information and high data throughput. Unlike typical SaaS applications, Void supports desktop-like interactions between applications, and allows the user to create or manipulate composite applications.

A surrogate, as developed above, can be used to construct user applications in the Cloud, in our original Void model. Fig. 3.2 shows the surrogate-based architecture of Void. Void not only hides the complexities and the diversities of different services, but also provides a unified and user-specific context to the user. From the users' perspective, the surrogate is their specific gateway for receiving various services including unified communications, content-specific services (e.g. search, add, mash-up, etc.) and other fundamental services (e.g. storage). Beyond that, the surrogate is home to their own customized XaaS applications or deploy user-specific content.

Benefits of Void:

- The software run by the user is streamlined by pushing complexity, compatibility and integration issues into the Edge Cloud.
- New services could be constructed through mash-up from existing elementary services.
- An application may depend on and require secure and efficient access to other services, e.g., the P2P engine depends on the index engine for locating a requested content.
- In the enterprise environment, depending on role, position and hierarchy, a user should be able to use a specific set of services (which are delivered through XaaS). The surrogate could be used to maintain user-profiles which allow administrators to add new services or delete the existing ones from the user's portal without modifying the XaaS layer.
- The XaaS provider could deploy the surrogates and new value added services. The present XaaS models only focus on enterprise–centric services. However, there also exists a large market for personal services.

VIII CONCLUSION

The usual dichotomy is client-based integration or server-based integration. The void has the added benefit of involving the support of the local ISP, which facilitates the integration of key functionalities such as security, mobility and QoS. This new dichotomy between the user and cloud based services, expands the range of applications that can be migrated away from the user's computer, while avoiding inconveniences of loss of control or data replication over the currently prevailing SaaS model.

REFERENCES:

[1] J. Broberga, R. Buyyaa, Z. Tarib, MetaCDN: harnessing 'storage clouds' for high performance content delivery, Journal of Network and Computer Applications 32 (5) (2009) 1012–1022.
[2] K. Beaty, A. Kochut, H. Shaikh, Desktop to cloud transformation planning, in: Proc. of the 2009 IEEE International Symposium on Parallel & Distributed Processing, 2009, pp. 1–8.

[3] X. Liu, et al. Towards service composition based on mashup, in: Proc. of the IEEE Congress on Services, 2007, pp. 332–339.
[4] D. Kovachev, R. Klamma, A cloud multimedia platform, in: Proc. of the 11th International Workshop of the Multimedia Metadata Community on Interoperable Social Multimedia Applications, 2010, pp. 61–64.
[5] S. Islam, J.-Ch. Grégoire, Network edge intelligence for the emerging nextgeneration Internet, Future Internet 2 (4) (2010) 603–623.
[6] Apache Tomcat. http://tomcat.apache.org/.
[7] Wikipedia. http://www.wikipedia.org/
[8] Java Media Framework (JMF). http://java.sun.com/javase/technologies/desktop/media/jmf/index.jsp.
[9] SailFin project. https://sailfin.dev.java.net/.
[10] GlassFish-open source application server. https://glassfish.dev.java.net/.
[11] The anue systems. http://www.anuesystems.com/
[12] Amazon Elastic Compute Cloud (amazon ec2). http://aws.amazon.com/ec2/.

M. Ramya Catherine is a post graduate scholar pursuing her master degree in Engineering in Information Technology, with Multimedia Technology as her specialization, in Karunya University, India.
E. Bijolin Edwin is an Assistant Professor in the department of Information Technology, Karunya University, India. His research interests are in Image Processing, Cloud Computing and Networking.