# Cloud Computing: The Future of Cost Effective and Enterprise Friendly Computing

Mohammad Ahmad Mohammad Nasimuddin
Department of Computer Science
College of Science & Arts, Sajar Campus
Shaqra University, Saudi Arabia

Nayyar Ahmed Khan
Department of Computer Science
College of Computing and IT, Shaqra Campus
Shaqra University, Saudi Arabia

***Abstract*—The paper is a brief introduction of the cloud computing perspective for the coming future in which the enterprise level computing is set to a very small cost-effective computation in which the cost of establishment is reduced by a huge factor. In the paper the basic introduction of the Cloud Computing, modals, characteristics, architecture, elements, security, features, technologies, difference between cloud and grid, problems in cloud, issues, difficulties, effects, objectives, essentials, and some case studies on cloud computing is given. The case study includes Blue Cloud from IBM and EC2 by Amazon is given. A case Study from Force.com is highlighted with the complete implementation of PaaS Architecture on Recruitment Cloud generated and managed by the Cloud Computing Environment. The conclusion is highlighted which yields a much more secure and cost-effective paradigm called as Cloud for effective computation and a distributed application based on a very small investment from large enterprises thereby reducing the bulk cost and pricing of development and maintenance cost.**

***Keywords— Cloud Computing, Amazon, Salesforce, EC2, IBM Bluemix***

## I. INTRODUCTION

Cloud Computing and its Features: "What I want is a robust, high performance virtual relational database that runs transparently over a cluster, nodes dropping in and out of service at will, read-write replication and data migration all done automatically. I want to be able to install a database on a server cloud and use it like it was all running on one machine." --Greg Linden's blog, 2006 Providing services on virtual machines allocated on top of a large physical machine pool A method to address scalability and availability concerns for large scale applications *Cloud computing* is a term used to describe both a platform and type of application. A cloud computing platform dynamically provisions, configures, reconfigures, and deprovisions servers as needed. Servers in the cloud can be physical machines or virtual machines. Advanced clouds typically include other computing resources such as storage area networks (SANs), network equipment, firewall and other security devices. Cloud computing also describes applications that data are extended to be accessible through the Internet. These *cloud applications* use large data centers and powerful servers that host Web applications and Web services. Anyone with a suitable Internet connection and a standard browser can access a cloud application [1][2]. A cloud is a pool of virtualized computer resources, which can perform the following task as:

1. Host a variety of different workloads, including batch-style back-end jobs and interactive, user facing applications
2. Allow workloads to be deployed and scaled-out quickly through the rapid provisioning of virtual machines or physical machines
3. Support redundant, self-recovering, highly scalable programming models that allow workloads to recover from many unavoidable hardware/software failures
4. Monitor resource use in real time to enable rebalancing of allocations when needed. [3]

1. 1 Modals Used for Cloud Computing:
A computing style in which the IT enabled services are given as a service on the network and not the service as a whole. It is more a specialization concept for the complete generalized mode of working for the system [4].

1. Acquisition Modal
   a. Service based modal: Only the result is needed and not the implementation logic.
2. Business Modal
   a. Usage Modal: Pay for what we want to use and pay for as much as we use.
3. Access Modal
   a. Connectivity Modal: Access services wherever we want and with any device
4. Technical Modal

1.1.1 Characteristics [5]:

1. Self Service and rich user experience
2. Increased System usage
3. Automated request and fulfillment
4. Rapid service
5. Massive scaling of services
6. Utility based usage matrices
7. Modular services based on domain
8. Shared services In-trusted domains
9. Reduced investments
10. Cost Efficiency

1.1.2 What empowers the Cloud:

1. Data Intensive applications.
2. Data center pressures.
3. Increased network capability and availability.
4. Shared services in business domains.
5. Rising energy cost.
6. Innovation and collaboration.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 12, December - 2017

1.1.3 Software as a service [6]:

1. Computer applications accessed directly over the internet.
2. No installation on local desktops or data center servers.
3. Applications available on demand on a subscription basis.
4. Documents can be recalled, shared or worked on from any computing device.

Examples:

- Online Project management tools Clarizen.
- Human resource applications Salesforce, Employ ease, Zoho.
- IBM Bluehouse – Web based social networking and collaboration service for business.
- Online office applications Google Docs, Blist, Slide Rocket.

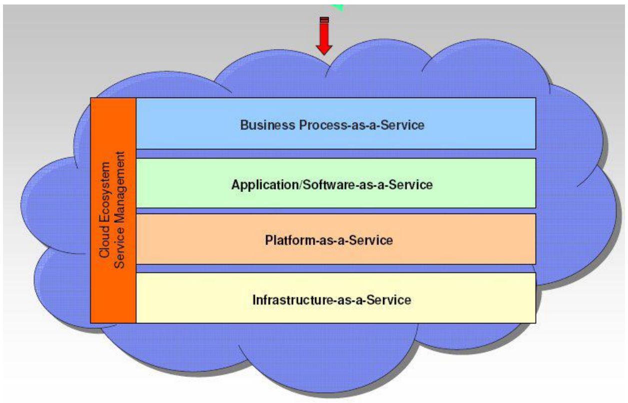



Fig 1. CLOUD SERVICE ARCHITECTURE SCHEMA

1.1.4 Cloud delivery modal:

1. Enterprise Internal Cloud: In this mode Security Clouds services are delivered behind a firewall and safe transaction takes place. [7]
2. Public Cloud: Cloud Services delivered over the internet.
3. Hybrid Cloud: Combination of above two services.

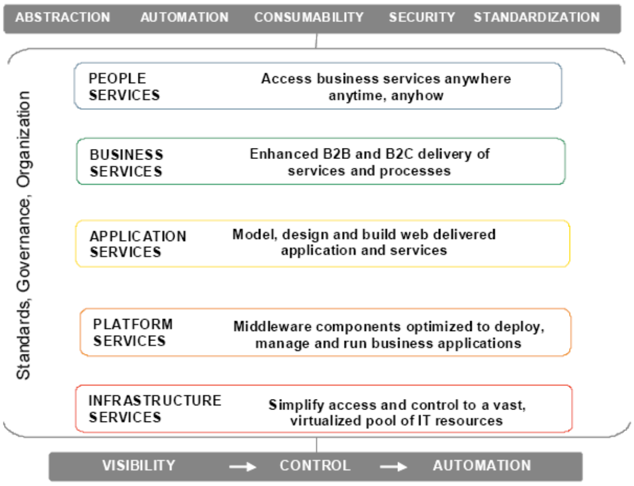



Fig 2. Cloud Models for Services

1.1.5 Elements in the cloud computing environment [8]

1. Service Creator
2. Service Catalogue Manager
3. Service Subscriber
4. Service Delivery Manager
5. Administrator
6. Service Consumer
7. Definition/Offering/Subscriptio/Production/Consumption/Termination

1.2 Security in Cloud Computing [9]

1. Authentication and role-based access control
   a. Federated Identity
   b. Single sign-on
2. Isolation Management
   a. Server
   b. Storage
   c. Network
3. Security for Image Management
   a. Security Metadata
   b. Access Control
   c. Authorization
4. Integrity management
   a. Virtual Image
   b. Integrity
5. Risk and Compliance
   a. Auditing and Configuration Management
   b. Enterprise-level Regulatory Compliance
6. Policy Management
7. Threat Management

1.3 Clouds Created and implemented by IBM [10]

1. Service Management for Cloud Computing
2. Tivoli Service Automation Management (TSAM)
3. Self-enablement Portal
4. Virtual Infrastructure Access
5. Scale out File Services
6. IBM security solutions for cloud computing
7. Virtual workplace continuity

1.3.1 Competitive Features [11] *

1. Zero capital outlay
2. No lock-ins
3. Try and buy
4. Autonomics
5. Support
6. Dynamic horizontal(global) and vertical(resource) scalability
7. No infrastructure hardware or software to maintain
8. Flexible migration strategies

1.3.2 Cloud Technologies [12] **

1. VMWare
2. Xen
3. 3Tera
4. Data Synapse
5. Amazon AWS
6. Hadoop

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 12, December - 2017

1.4 Cloud as Represented by the service architectures:

1.4.1 Cloud Software as a Service (SaaS) [13]

Features of SaaS Applications:

Scalable - Handle growing amounts of work in a graceful. Manner Multi-tenancy - One application instance may be serving hundreds of companies Opposite of multi-instance where each customer is provisioned their own server running one instance Metadata driven configurability Instead of customizing the application for a customer (requiring code changes), one allows the user to configure the application through metadata

1.4.2 Cloud Platform as a Service (PaaS)

Deploy customer-created applications to a cloud

1.4.3 Cloud Infrastructure as a Service (IaaS)

Rent processing, storage, network capacity, and other fundamental computing resources. In today's global competitive market, companies must innovate and get the most from its resources to succeed. This requires enabling its employees, business partners, and users with the platforms and collaboration tools that promote innovation. Cloud computing infrastructures are next generation platforms that can provide tremendous value to companies of any size. They can help companies achieve more efficient use of their IT hardware and software investments and provide a means to accelerate the adoption of innovations [14]. Cloud computing increases profitability by improving resource utilization. Costs are driven down by delivering appropriate resources only for the time those resources are needed. Cloud computing has enabled teams and organizations to streamline lengthy procurement processes. Cloud computing enables

Innovation by alleviating the need of innovators to find resources to develop, tests, and make their innovations available to the user community [15]. Innovators are free to focus on the innovation rather than the logistics of finding and managing resources that enable the innovation.

1.5 BENEFITS OF CLOUD COMPUTING FOR THIN CLIENTS

1. With traditional desktop computing, we run copies of software programs on our own computer. The documents we create are stored on our own pc [16].
2. Although documents can be accessed from other computers on the network, they can't be accessed by computers outside the network. This is PC-centric.
3. With cloud computing, the software programs one use isn't run from one's personal computer, but are rather stored on servers accessed via the Internet.
4. If a computer crashes, the software is still available for others to use. Same goes for the documents one creates; they're stored on a collection of servers accessed via the Internet.
5. Anyone with permission can not only access the documents, but can also edit and collaborate on those documents in real time. Unlike traditional computing, this cloud computing model isn't PC-centric, it's document-centric [17].

1.6 Differences between cloud and grid computing:

Problems with the current scenario (Statically):

| | EGEE Grid | Amazon Cloud |
|---|---|---|
| Target Group | Scientific community | Business |
| Service | short-lived batch-style processing (job execution) | long-lived services based on hardware virtualization |
| SLA | Local (between the EGEE project and the resource providers) | Global (between Amazon and users) |
| User Interface | High-level interfaces | HTTP(S), REST, SOAP, Java API, BitTorrent |
| Resource-side middleware | Open Source (Apache 2.0) | Proprietary |
| Ease of Use | Heavy | Light |
| Ease of Deployment | Heavy | Unknown |
| Resource Management | probably similar | |
| Funding Model | Publicly funded | Commercial |

Summary of „An EGEE Comparative Study: Grids and Clouds Evolution or Revolution" by Markus Klems

Fig 3. Comparison Chart for Grid and Cloud Computing

1.6.1 Problems with the current scenario (Statically) [18]:

1. 11.8 million servers in data centers
2. Servers are used at only 15% of their capacity
3. 800 billion dollars spent yearly on purchasing and maintaining enterprise software
4. 80% of enterprise software expenditure is on installation and maintenance of software
5. Data centers typically consume up to 100 times more per square foot than a typical office building
6. Average power consumption per server quadrupled from 2001 to 2006.
7. Number of servers doubled from 2001 to 2006
8. Standard 9000 square foot costs $21.3 million to build with $1 million in electricity costs/year
9. Data centers consume 1.5% of our Nation's electricity (EPA)
10. .6% worldwide in 2000 and 1% in 2005
11. Green technologies can reduce energy costs by 50%
12. IT produces 2% of global carbon dioxide emissions Cloud Economics:
13. Estimates vary widely on possible cost savings
14. "If you move your data center to a cloud provider, it will cost a tenth of the cost." – Brian Gamage, Gartner Fellow
15. Use of cloud applications can reduce costs from 50% to 90% - CTO of Washington D.C.
16. IT resource subscription pilot saw 28% cost savings - Alchemy Plus cloud (backing from Microsoft)

1.6.2 Cloud Computing is breaking out in the scenario [19]:

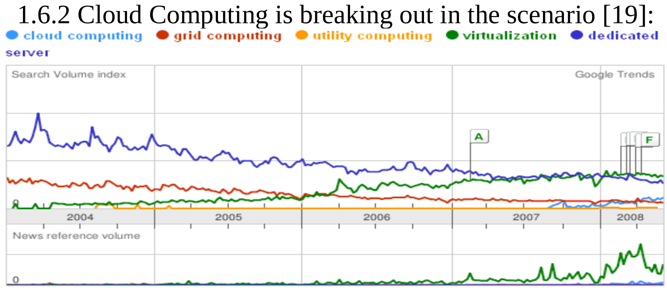



Fig 4. Cloud vs other technologies.

1.6.3 Difficulties in Cloud Computing [20]:
- Continuous high availability
- Consistency
- Interoperability and standardization
- Scalability of all components
- Data secrecy
- Legal and political problem of data store and translation across regions
- Performance issue
- Difficulty customizing

1.6.4 Issues in Cloud Computing [21]:
- Privileged user access.
- Regulatory compliance.
- Data location.
- Data segregation.
- Recovery.
- Investigative support.
- Long-term viability.

1.6.5 Effects of Cloud Computing:
1. Small enterprises use public SaaS and public clouds and minimize growth of data centers
2. Large enterprise data centers may evolve to act as internal clouds
3. Large enterprises may use hybrid cloud infrastructure software to leverage both internal and public clouds
4. Public clouds may adopt standards in order to run workloads from competing hybrid cloud infrastructures

1.6.6 Cloud Security Features [22]:
1. Data Fragmentation and Dispersal
2. Dedicated Security Team
3. Greater Investment in Security Infrastructure
4. Fault Tolerance and Reliability
5. Greater Resiliency
6. Hyper visor Protection Against Network Attacks
7. Possible Reduction of C&A Activities (Access
8. to Pre-Accredited Clouds)
9. Simplification of Compliance Analysis
10. Data Held by Unbiased Party (cloud vendor assertion)
11. Low-Cost Disaster Recovery and Data Storage Solutions
12. On-Demand Security Controls
13. Real-Time Detection of System Tampering
14. Rapid Re-Constitution of Services
15. Advanced Honeynet Capabilities

1.7 What are the objectives of Cloud Computing [23]:

Core objectives and principles that cloud computing must meet to be successful:
1. Security
2. Scalability
3. Availability
4. Performance
5. Cost-effective
6. Acquire resources on demand
7. Release resources when no longer needed
8. Pay for what you use
9. Leverage others' core competencies
10. Turn fixed cost into variable cost.

## II. A Case Study from Force.Com (PaaS) Applications for Cloud Computing Environment

### *A. The Basics of an App*
1. Tabs
2. Forms
3. Links

2.1 The Benefits of a Force Platform App
1. Data Centric: Application based on the structure information such as you might find in a database or an XML file. We can find these data centric apps everywhere, in small desktop databases like Microsoft Access or FileMaker, all the way to the huge systems running on database management systems like Oracle or MySQL [24]

2. Collaborative Applications: A collaborative app is an application with data and services that are shared by multiple users in different locations. Unlike more traditional forms of software that are installed on a single machine and are hard to access from a distance, collaborative apps on the platform can be accessed from anywhere in the world with only a Web browser.

2.2 A Metadata-Driven Development Model: Force Platform also uses a *metadata-driven development model* to help app developers become more productive in putting together basic apps. It means that the basic functionality of an app—that is, the tabs, forms, and links— is defined as metadata in a database rather than being hard-coded in a programming language. When a user accesses an app through the Force Platform, [25] it renders the app's metadata into the interface the user experiences. Developers who want to programmatically manage their app's metadata can use the *Force Platform Metadata Application Programming Interface (API)*. The Force Platform Metadata API provides an alternative to the platform's user interface by allowing developers to directly modify the XML files that control their organization's metadata. Developers can also use the Force Platform Metadata API to migrate configuration changes between organizations, and create their own tools for managing organization and application metadata. For information on the Force Platform Metadata API, see http://www.salesforce.com/us/developer/docs/api_meta/index.htm [26]

2.2.1 The Force Platform Web Services API: The platform's metadata-driven development model allows app developers to quickly build a lot of *native functionality*—that is, functionality that's understood by the platform and that can be defined in metadata. However, sometimes app developers need an app to do something that isn't available natively on the platform. They want to go beyond the metadata model and use traditional programming to create behaviors that fall outside of the platform's constraints. To do this, they can use the *Force Platform Web Services API.*

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 12, December - 2017

2.2.2 Apex
As you might expect from the company that delivered the world's first cloud computing platform, salesforce.com also introduced the world’s first cloud computing programming language, Apex. Apex runs on Force Platform servers, and is specifically designed for building business applications to manage data and processes within the larger context of the Force Platform. The language provides a uniquely powerful and productive approach to creating functionality and logic. [27]
http://wiki.apexdevnet.com/index.php/Apex_Code:_The_World's_First_On-Demand_Programming_Language.

2.2.3 Visual force: At the front of any great business application is a great user interface that's easy to use, powerful, and suited exactly for the tasks, users, and devices the application serves. Visual force is a complete framework for creating such user interfaces, enabling any kind of interface design and interaction to be built and delivered entirely in the cloud. The user interfaces you build with Visual force can extend the standard Force Platform look and feel, or replace it with a completely unique style and set of sophisticated interactions. Because Visual force markup is Ultimately rendered into HTML, designers can use Visual force tags alongside standard HTML, JavaScript, Flash, or any other code that can execute within an HTML page on the platform. [28]

2.2.4 The AppExchange Directory: The final piece of technology that differentiates the Force Platform from other platforms is the *AppExchange*. The AppExchange is a Web directory where apps built on the Force Platform are available to salesforce.com customers to browse, demo, review, and install.

2.3 Recruiting App for “Universal Containers” a firm:
Universal Containers needs a centralized application that can bring all of its recruiting and hiring processes together, and the company has hired us to solve this problem. Our approach will be to leverage their Sales force account and build a recruiting application on the Force Platform. We’re going to introduce Universal Containers to the world of cloud computing!

2.4 The application must be capable enough to perform following tasks [29]:
1. Track positions in all stages of the process, from those that are open to those that have been filled or canceled.
2. Track all of the candidates who apply for a particular position, including the status of their application (whether they've had a phone screen, are scheduled for interviews, have been rejected or hired, or have passed on an offer that was presented).
3. Track the posting of jobs on external employment websites, such as Monster.com
4. Allow employees to post reviews for candidates whom they've interviewed.
5. Provide security for the recruiting data so that it's not mistakenly viewed, edited, or deleted by employees who shouldn't have access.
6. Automatically inform the relevant recruiter about the next steps that should be taken when a decision has been made about an applicant.
7. Automatically inform all employees of new positions that have been posted.

## III. COMPONENTS IN THE APPLICATIONS

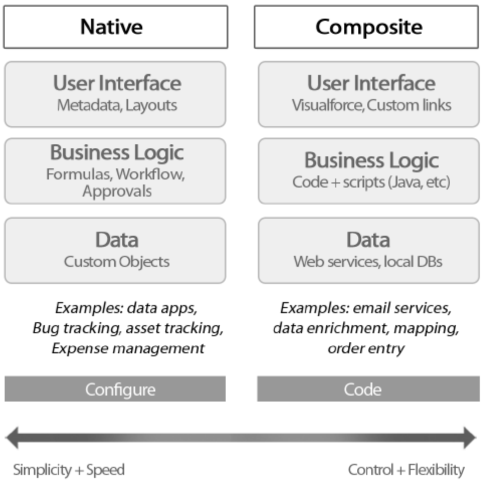
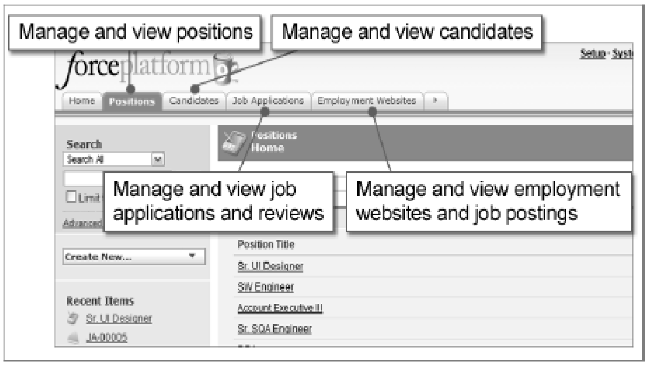



Fig 5. Custom Objects

Custom objects are the native components that model the data we need to store in our Recruiting App. Similar to a database table, a custom object is composed of several fields that store information such as a job applicant's name, or the maximum salary for a particular position [30].

1. Position
2. Candidate
3. Job Application
4. Review
5. Job Posting
6. Employment Website

Three of these objects, Candidate, Position, and Job Application, will be displayed as tabs in our application. When a user clicks one of the tabs, he or she will have access to individual instances of that particular object, as shown in the following screenshot [31]



International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 12, December - 2017

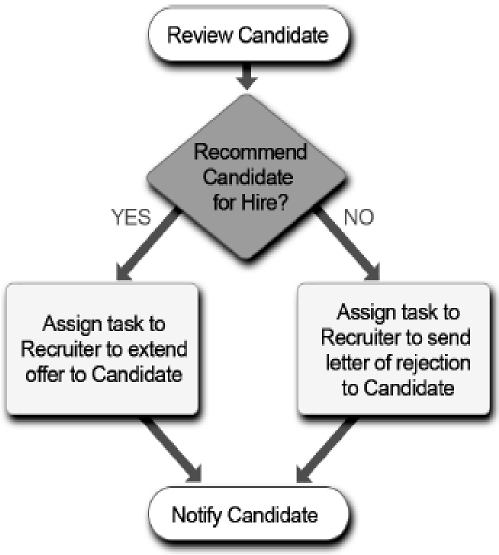



Fig 6. WORKFLOW WHEN A JOB APPLICATION STATUS IS CHANGED:

3.1 DATABASE CONCEPT:

1. A *database* is an organized collection of Information.
2. A database table stores information about a single type of person, thing, or concept—such as a job position. In the Force Platform, we use the term *object* here (even though an object is much more than this, as you'll see) [32].
3. A database row, or *record* in Force Platform terms, represents a single instance of an object such as the SW Engineer position.
4. A *field* stores a particular piece of information on a record.
5. *Relationships* define the connection between two objects, and objects are related to each other through the use of common fields.

3.2 SECURING AND SHARING OF DATA:
The Key Issue in Cloud Computing [33]

1. Object-Level Security
   The bluntest way that we can control data is by preventing a user from seeing, creating, editing, and/or deleting any instance of a particular type of object, like a position or review. Object-level access allows us to hide whole tabs and objects from particular users, so that they don't even know that type of data exists.
2. Field-Level Security
   A variation on object-level access is field level access, in which a user can be prevented from seeing, editing, and/or deleting the value for a particular field on an object. Field-level access allows us to hide sensitive information like he maximum salary for a position or a candidate's social security number without having to hide the whole object.
3. Record-Level Security
   To control data with a little more finesse, we can allow particular users to view an object, but then restrict the individual object records that they're allowed to see.

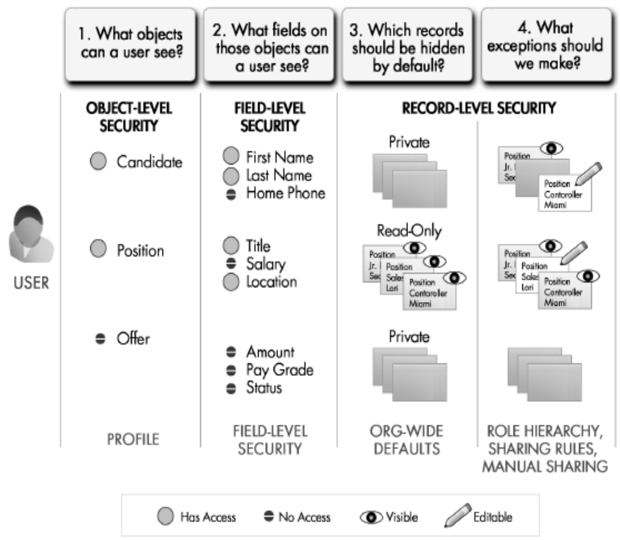



Fig 7. Using Custom Workflow and Approval Processes

Workflow is a Force Platform business-logic engine that allows us to automatically send email alerts, assign tasks, or update field values based on rules that we define. Any time that changes to a record meet the conditions in a workflow rule, the platform automatically performs any actions associated with the rule.

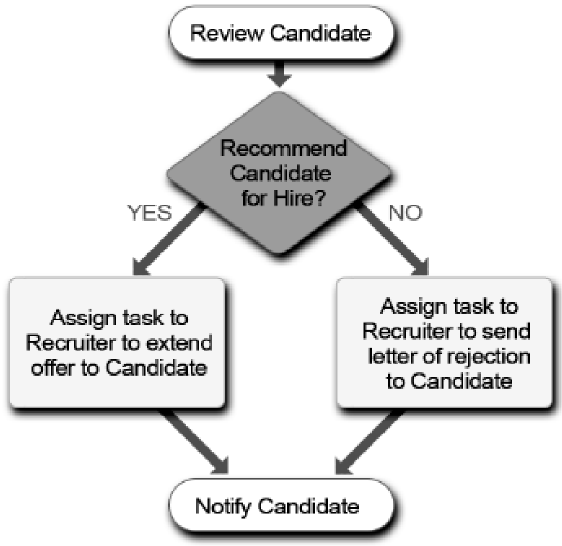



Fig 8. Notification Workflow

*3.3 Workflow Tasks*

A *workflow task* assigns a task to a user according to a particular template. Just as in Microsoft Outlook, tasks include information about something that needs to be done by a certain time, such as making a telephone call or returning a library book [34].

*3.4 Workflow Field Updates*

A *workflow field update* changes the value of a particular field on the record that initially triggered the workflow rule.

### *3.5 Workflow Email Alerts*

A *workflow email alert* sends an email according to a specified email template. Unlike workflow tasks, which can only be assigned to users of the app, workflow alerts can be sent to any user or contact, as long as they have a valid email address.

### *3.6 Analyzing Data with Reports and Dashboards*

Reports are summaries of the data that's stored in an app. They consist primarily of a table of data, but can also include data filters, groupings, and a customized graph [35].

Report Formats

1. *Tabular reports* are the simplest and fastest way to look at your data. Similar to a spreadsheet, they consist simply of an ordered set of fields in columns, with each matching record listed in a row. While easy to set up, they can't be used to create groups of data or graphs.
2. *Summary reports* are similar to tabular reports, except that they also allow you to group rows of data, view subtotals, and create graphs.

*Matrix reports* are the most complex kind of report available, allowing you to group records both by row and by column.

## IV. CONCLUSION

This paper gives a review of all methodologies being used in Cloud Computing methods based on contents and also about its classification techniques. Along with it this paper also adds up the challenges available in every technique or method which the reader can take it as their research area trying to figure out the solution for the problems or challenges. The case study of Force.com can also be an additional benefit for the researchers going ahead in the field of Cloud Computing that can act as a mode of very simple and cost-effective solution for the system design and dynamics

## REFERENCES

[1]. Cloudera, Hadoop training and support [online]. Available from: http://www.cloudera.com/.
[2]. TC3 Health Case Study: Amazon Web Services [online]. Available from: http://aws.amazon.com/solutions/ case-studies/tc3-health/.
[3]. Washington Post Case Study: Amazon Web Services [online]. Available from: http://aws.amazon.com/solutions/case-studies/washington-post/.
[4]. Amazon.com CEO Jeff Bezos on Animoto [online]. April 2008. Available from:
[5]. http://blog.animoto.com/2008/04/21/amazon-ceo-jeff-bezos-on-animoto/.
[6]. Black Friday traffic takes down Sears.com.Associated Press (November 2008).
[7]. ABRAMSON, D., BUYYA, R., AND GIDDY, J. A computational economy for grid computing and its implementation in the Nimrod-G resource broker. Future Generation Computer Systems 18, 8 (2002), 1061–1074.
[8]. ADMINISTRATION, E. I. State Electricity Prices, 2006 [online]. Available from:
[9]. http://www.eia.doe.gov/neic/rankings/stateelectricityprice.htm.
[10]. AMAZON AWS. Public Data Sets on AWS [online]. 2008. Available from: http://aws.amazon.com/publicdatasets/.
[11]. BARROSO, L. A., AND HOLZLE, U. The Case for Energy-Proportional Computing. IEEE Computer 40, 12 (December 2007).
[12]. BECHTOLSHEIM, A. Cloud Computing and Cloud Networking. talk at UC Berkeley, December 2008.
[13]. BIALECKI, A., CAFARELLA, M., CUTTING, D., AND O'MALLEY, O. Hadoop: a framework for running applications on large clusters built of commodity hardware. Wiki at http://lucene. apache. org/hadoop.
[14]. BRODKIN, J. Loss of customer data spurs closure of online storage service 'The Linkup'. Network World (August 2008).
[15]. CARR, N. Rough Type [online]. 2008. Available from: http://www.roughtype.com.
[16]. CHANG, F., DEAN, J., GHEMAWAT, S., HSIEH, W., WALLACH, D., BURROWS, M., CHANDRA, T., FIKES, A., AND GRUBER, R. Bigtable: A distributed storage system for structured data. In Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI'06) (2006).
[17]. CHENG, D. PaaS-on omics: A CIO's Guide to using Platform-as-a-Service to Lower Costs of Application Initiatives While Improving the Business Value of IT. Tech. rep., Long Jump, 2008.
[18]. DEAN, J., AND GHEMAWAT, S. MapReduce: simplified data processing on large clusters. In OSDI'04: Proceedings of the 6th conference on Symposium on Operating Systems Design & Implementation (Berkeley, CA, USA, 2004), USENIX
[19]. www.rgrossman.com
[20]. L. Klein rock. A vision for the Internet. ST Journal of Research, 2(1):4-5, Nov. 2005.
[21]. S. London. INSIDE TRACK: The high-tech rebels. Financial Times, 06 Sept. 2002. http://search.ft.com/nonFtArticle?id=020906000371 [18 July 2008]
[22]. I. Foster and C. Eshelman (eds). The Grid: Blueprint for a Future Computing Infrastructure. Morgan Kaufmann, San Francisco, USA, 1999.
[23]. M. Chetty and R. Buyya. Weaving Computational Grids: How Analogous Are They with Electrical Grids? Computing in Science and Engineering, 4(4):61–71, July–Aug. 2002.
[24]. A. Weiss. Computing in the Clouds. networker, 11(4):16-25, Dec. 2007.
[25]. Twenty Experts Define Cloud Computing, http://cloudcomputing.syscon.com/read/612375_p.htm [18 July 2008].
[26]. R. Buyya, D. Abramson, and S. Venugopal. The Grid Economy. Proceedings of the IEEE, 93(3): 698-714, IEEE Press, USA, March 2005.
[27]. S. Venugopal, X. Chu, and R. Buyya. A Negotiation Mechanism for Advance Resource Reservation using the Alternate Offers Protocol. In Proceedings of the 16th International Workshop on Quality of Service (IWQoS 2008), Twente, The Netherlands, June 2008.
[28]. D. Hamilton. 'Cloud computing' seen as next wave for technology investors. Financial Post, 04June2008.http://www.financialpost.com/money/story.html?id=562877 [18 July 2008]
[29]. Morgan Stanley. Technology Trends. 12 June 2008. http://www.morganstanley.com/institutional/techresearch/pdfs/TechTrends062008.pdf [18 July2008]
[30]. K. Keahey, I. Foster, T. Freeman, and X. Zhang. Virtual workspaces: Achieving quality of service and quality of life in the Grid. Scientific Programming, 13(4):265-275, October 2005.
[31]. I. Llorente, OpenNebula Project. http://www.opennebula.org/ [23 July 2008]
[32]. Amazon Elastic Compute Cloud (EC2), http://www.amazon.com/ec2/ [18 July 2008]
[33]. Google App Engine, http://appengine.google.com [18 July 2008]
[34]. Microsoft Live Mesh, http://www.mesh.com[18 July 2008]
[35]. Sun network.com (Sun Grid), http://www.network.com [18 July 2008]
[36]. X. Chu, K. Nadiminti, C. Jin, S. Venugopal, and R. Buyya. Aneka: Next-Generation Enterprise Grid Platform for e-Science and e-Business Applications. In Proceedings of the 3th IEEE International Conference on e-Science and Grid Computing (e-Science 2007), Bangalore, India, Dec. 2007.
[37]. Y. Fu, J. Chase, B. Chun, S. Schwab, and A. Vahdat. SHARP: an architecture for secure resource peering. ACM SIGOPS Operating