User Centric Smart Services on the Cloud

Karuna P. Joshi[1], Yelena Yesha[1], Ant A. Ozok[2] , Yaacov Yesha[1] , Ashwini Lahane[1]
Hari Kalva[3], Ankur Agarwal[3], and Borko Furht[3]

[1] Computer Science and Electrical Engineering Department,
[2] Information Systems Department,
University of Maryland Baltimore County, MD, USA
{kjoshi1, yeyesha, ozok, yayesha, alahane1}@umbc.edu

[3] Computer & Electrical Engineering and Computer Science Department,
Florida Atlantic University, Boca Raton, FL, USA
{hari, ankur, borko}@cse.fau.edu

Abstract. In this chapter we describe our vision for the next generation of IT services. Services will be automatically discovered, procured and integrated with the service consumer's technical environment. This whole process will be determined by the policies defined by the consumer and will be transparent to the consumer. We also define the new applications of Policy Manager, Service Manager, Service Procurer and Service Integrator that will allow Smart Services to operate efficiently on the Smart Internet. Usability and User issues are also identified. We illustrate the Smart Services with examples from the health care and multimedia domain.

Keywords: Smart Internet, Smart Services, Services automation.

1 Introduction

It is increasingly being recognized that the Internet in its current state is server-centric where the content, control and performance lies with the servers running the websites. The onus is on the consumer to search for the required data/information across various websites and combine it as needed. The next generation of the Internet, termed Smart Internet [2], advocates a user-centric model for the web instead of the server-centric model that exists today. The Smart Internet addresses the technological gaps due to lack of integration, individualization, user control, collaborative services that exist in the current rendition of the Internet.

In this chapter, we articulate a similar vision for services, specially virtualized services on the cloud. Virtualized service models are emerging and IT development and maintenance which was previously either in-house or outsourced is being replaced by this new delivery model where businesses purchase IT components, like software, hardware or network bandwidth, as services from providers distributed globally. In such scenarios, multiple providers often collaborate to create a single service for an organization. In some cases, organizations utilize multiple service providers to mitigate risks associated with a single provider. In other cases, they may utilize a single provider who in turn utilizes the services of other providers. Moreover, the component service of a provider could simultaneously participate in several composed service orchestrations. This model has also been termed Service Virtualization [1]. Virtualization implies that the service with which an end customer interacts may be composed of many others, and each service in turn could depend on backend applications (database, web server) and resources (storage, bandwidth, CPU). It could be delivered remotely to the consumer via a computing grid or cloud. The service, in effect, is virtualized on the cloud. This virtualized model of service delivery potentially allows easier service customization, better resource utilization and greater responsiveness on part of the service providers. In this model, the service is acquired through the "on demand" pull technology. This is true of services that are purely IT in nature (e.g. Software as a Service (SaaS) or Infrastructure as a Service (IaaS)), as well as services that are IT enabled (ITeS) but involve human contact (e.g. those provided via contact centers). It is possible in these scenarios that neither the hardware infrastructure, nor the software; and not even the people running these services belong to the organization that uses the service.

Creation of a user-centric model for virtualized services presents new challenges. While the concept of 'Service on demand', i.e. a user requesting service or its components when needed, is promising; there appears to be no rush from organizations to adopt this new model. One of the key barriers is the lack of infrastructure to enable automatic management, procurement and integration of services. Smart Internet is essential to realize the full benefit of the virtualized delivery model. In this paper we describe our vision of how services will be more automatically managed in a Smart Internet environment. Another barrier is the Usability and User related issues of on-demand services. In this chapter we also look into this in detail and identify the open issues that will need to be addressed in the Smart Internet. We have provided examples from multimedia and health care domain to illustrate our vision. We review related work in this area in section 2 and detail the proposed service flow in section 3. We look into the Usability issues that will arise for such services in section 4. Section 5 and 6 provides some examples from the healthcare and multimedia domain respectively and list the open issues that will need to be addressed for services on the Smart Internet. We conclude in section 7 and provide an overview of our ongoing work

2 Related Work

A user-centric model of the Internet mandates the capability of individualizing or personalizing websites for each user based on their preferences, traversal patterns, functional domain, services consumed etc. Personalization of web has been extensively researched. Researchers have tried to enhance it by improving upon Internet's searching capabilities. Joshi and Jiang [13] had proposed an algorithm to cluster search results to provide users with a personalized view of their web queries. Stanford NLP group [14] have developed PageRank algorithm to decide the ranking of web pages returned after a search. More recently, Balke and Wagner [14] have proposed an algorithm featuring the expansion of service requests by user-specific demands to enable personalized selection of web services. However, most of these approaches have been based on the current server-centric model on the Internet.

Researchers have also concentrated on developing methodologies for virtualized services. Papazoglou and Van Den Heuvel [3] have proposed a methodology for developing and deploying web services using service oriented architectures. Their approach, however, is limited to the creation and deployment of web services and does not account for virtualized environment where services are composed on demand. Providers may need to combine their services with other resources or providers' services to meet consumer needs. Other methodologies, like that proposed by Bianchini et al. [4], do not provide this flexibility and are limited to cases where a single service provider provides one service. Zeng et al. [5] address the quality based selection of composite services via a global planning approach but do not cover the human factors in quality metrics used for selecting the components. Maximilien and Singh [6] propose an ontology to capture quality of a web service so that quality attributes can be used while selecting a service. Their ontology is limited by the fact that it considers single web services, rather than service compositions. Black et al. [7] have proposed an integrated model for IT service management. Their model is limited to managing the service from the service provider's perspective. Milanovic et al. [30] have summarized the key issues in web services composition. Dustdar et. al [29] have presented several different composition strategies and have reviewed the dynamic web service composition approach which is relevant to the on-demand service composition.

In a virtualized service-oriented environment, consumers and providers need to be able to exchange information, queries, and requests pertaining to the data and policies with some assurance that they share a common meaning. One possible approach to this issue is to employ Semantic Web techniques for modeling and reasoning about services related information. Semantic Web is an enhancement of the World Wide Web that deals primarily with data instead of documents. It enables data to be annotated with machine understandable metadata, allowing the automation of their retrieval and their usage in correct contexts. Semantic Web technologies include languages such as Resource Description Framework (RDF) [10] and Web Ontology Language (OWL) [9] for defining ontologies and describing metadata using these ontologies as well as tools for reasoning over these descriptions. Web Services Description Language (WSDL) [8] is an XML based language that provides a model for describing web services. It defines services as collections of network endpoints, or ports operating on messages containing either document-oriented or procedure-oriented information. Business Process Execution Language (BPEL) [11] is a standard executable language for specifying business process behavior based on web services. Its messaging facilities depend on WSDL. Semantic web technologies can be used to provide common semantics of service information and policies enabling all agents who understand basic Semantic Web technologies to communicate and use each other's data and services effectively.

3 Services on Demand

Existing methodologies for designing and deploying web based services put the control in the hands of the service providers who plan and design the service, decide its composition, delivery mode and then wait for consumer request of the service. Services are mainly developed by using the semantic web technologies. While semantic web languages are easier to read compared to computer languages, like C or Java; they still require the consumers to possess technical expertise in semantic web languages to be able to consume the services.

We envision that with the development of the Smart Internet and further adoption of virtualized services and cloud computing models, the next generation of service lifecycle will be fully automated. It will be transparent to the consumer and services needed will be automatically identified by the service environment, discovered and procured in the cloud and seamlessly integrated back with the consumer's service environment. Figure 1 illustrates the flow of information in automated services. The consumer's services environment will consist of three new applications/toolsets that will completely automate how services are acquired, consumed and managed in a Smart Internet environment. The service environment will consist of a Service Manager, Policy Manager, Service Monitor and Service log that will help manage the services consumed in an organization. Service procurer will automatically acquire the desired services from the service cloud. Service Integrator will seamlessly integrate the acquired service into the existing service environment. We describe the applications and the lifecycle of smart services in detail below.

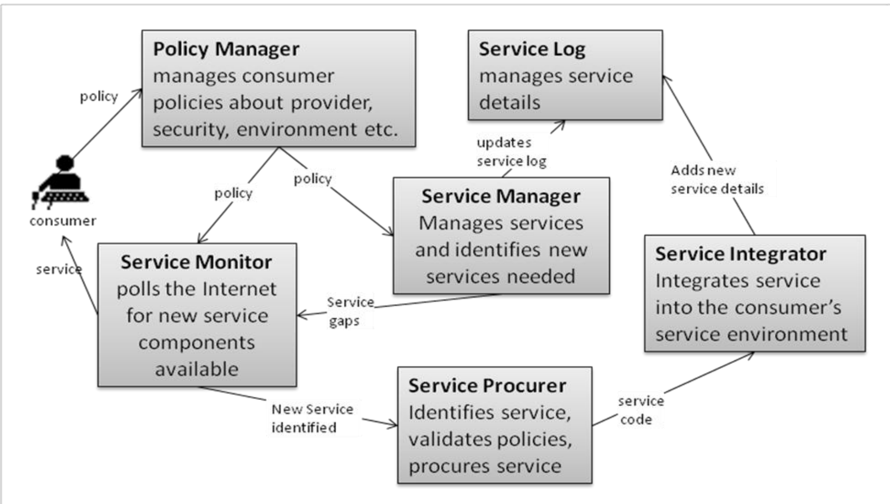

Figure 1: Smart services will consist of service manager, service monitor, service procurer and service integrator applications built into the service infrastructure.

3.1 Smart Services Consumer Environment

Current design of the Internet requires the consumers to manually search/discover and procure a service available on the web and then integrate it within their IT environment. This approach is cumbersome for the consumers and also prone to guess work on the user's part as to which service is needed. Smart Internet in the future would enable automation of this step by dynamically identifying the services that a user needs and automatically procuring them. This would be similar to the existing upgrade utility on computers that automatically checks the software versions installed and then runs the update in the background after confirming with the user. For instance, if a user is consuming a service that provides financial information, then additional services that are either missing or could enhance the information displayed would be automatically identified and procured for the consumer depending on the service procurement policies that the consumer has defined. Smart services infrastructure running on the consumer's IT environment will include the applications listed below.

3.2 Service Manager

Service Manager manages the services by tracking their quality and performance. It also identifies gaps in the services and inform the Service Monitor of them so that the Monitor could search for services that can fill the gap. If a service does not meet the desired performance levels, the service manager will send alerts to humans or automatically terminate the service and send instructions to the service monitor to poll for another service.

Many software providers like Microsoft [19], IBM [20], Hewlett Packard [21], Oracle [21] etc. have released basic versions of service managers in the market. However these products are platform specific. For instance, the Microsoft service manager manages only Windows specific services and Oracle server manages only Oracle services. Due to this the consumer has to either manually integrate service management across various platforms or write a proprietary program to manage the service flow across platforms. We envision that in Smart Services, the Service Manager application will be platform independent and will provide a user-centric aggregated view of all the services in the consumer's IT environment irrespective of service provider.

Service Manager will also consist of a Service Log that will manage a list of all services in the consumer's service environment. It will track the service by its functionality, provider, dependent services/components and technical features. The service log will also act as an audit for each service by logging the number of times it was used, the user name and cost to the organization.

3.2 Service Monitor

The current state of art of the Internet requires that the consumer first identify the service that s/he wishes to use and then discover a provider on the cloud who will be able to compose the service to match their requirements. Smart services will consist of a Service Monitor program to regularly poll the cloud and identify any new services or service upgrades that have been released. Service Monitor will refer to the policies stored in the policy manager to determine the domains and constraints that it should filter off. For instance, if the consumer is using a document management service to manage their organizational files, then the service monitor will poll the service cloud for other similar services that might be available at reduced cost. This functionality is similar to the update application available on Windows environment which lets the users know as soon as a security patch is available to update their operating system. The service monitor will also protect the service environment from being inundated by sample services or advertisement spam sent by service providers.

When the service monitor identifies a service that needs to be procured, it sends the list of requirements along with the associated policies determining the service constraints to the Service Procurer.

3.3 Policy Manager

Policy Manager manages the policies decided by the consumer on service discovery and procurement along with the policies pertaining to business rules. For instance, the consumer may specify a limited list of providers for service discovery; or may want to limit service discovery to a geographical location. Similarly the consumer may specify the budgetary, security or language constraints as part of their procurement policy. The Service Policy Master would ideally contain all the business rules and IT policies of the organization that should be considered by the consumer before procuring a service.

The collection of consumer's policies is translated to a description using semantic web languages such as RDF and OWL. We plan to use the Rei policy framework [12] (http://rei.umbc.edu) to develop the policies. This will ensure that the Service Policy Master uses a standard interface, and replacing it does not require reentering the policies.

The collection of consumer's policies is translated to a description using semantic web languages such as RDF and OWL. Examples of policy frameworks that may be used are Web Services Policy Framework [28], that includes WS-Policy assertions and WS-Policy Attachments and the Rei policy framework [12] (http://rei.umbc.edu) to develop the policies. This will ensure that the Service Policy Master uses a standard framework and that policies can be easily reused and shared when this is desirable

3.4 Service Procurer

Service procurer searches for services on the cloud that match the specified service. The discovery is constrained by functional and technical attributes identified by the Service monitor, and also by the budgetary, security, data quality and agent policies of the consumer defined in the policy master. One approach that the service procurer could take for discovering services would be to run a query against the services registered with a central registry or governing body. Alternatively, it could limit the service search to a set of providers specified in the consumer's provider policy. While discovering the service, the procurer will need references for providers that match the requirements, especially if the provider is not a part of the preferred vendor list defined in the policy. Service procurer will be able to get certification for the service provider by contacting a central registry, like UDDI [18].

If the service procurer identifies the exact service matching the specifications and constraints then it automatically procures it and transfers the control to the service integrator. If the procurer discovers multiple providers for the service, each meeting the consumer's policies, then it creates an aggregated view of the search results providing instantaneous comparison of the services available, their cost, their constraints and the service gap. The control is then transferred to the consumer who finalizes the service provider. While the existing Internet infrastructure does not allow aggregation of website from multiple servers or transfer of control to the user, Smart Internet has been envisioned to enable this. After the consumer finalizes the service provider, service procurer begins contract negotiation with the provider. Service Level Agreements (SLAs) are finalized during this negotiation. SLAs define the service data, delivery mode, agent details, compliance policy, quality and cost of the service.

After the service procurer acquires the service, it transfers control to the service integrator.

3.5 Service Integrator

Once the service is procured, the next step is to integrate it into the IT environment that exists on the consumer's side. Currently web services require a skilled administrator at the consumer end to integrate the service with other existing applications. This adds to the overall cost as well as the maintenance of the service and is a major disincentive for organizations to adopt services technology.

We envision that in the Smart Internet, there will be a Service Integrator program that will automatically integrate the new service procured into the consumer's existing service infrastructure, thereby removing the need of manual integration. Often different services procured could be combined to form a single service. In other instances, the service procured could be integrated into multiple services. For instance, a currency converter service procured could be integrated with multiple financial application services. Either way, this integration would be transparent and seamless to the end user. Service integrator will also update the service manager, service monitor and the service log with the details of the new service procured.

4 Usability and User Issues

As the "on demand" service systems deal with users, human-computer interaction issues need to be taken into consideration in design, implementation, and maintenance phases of the system and beyond. In designing such software systems with a service focus, emphasis on usability and user issues may be critical for maintaining optimal user and satisfaction. Variations of usability design guidelines initially created by

Shneiderman [16] and Nielsen [17] can be adopted for service oriented systems and its components. Time-critical tasks can be run more effectively on systems with optimally usable interfaces that present the relevant content in a user friendly design.

The concept of “smart services” may bring with it a number of issues that previously may not have been noted by researchers. It should be noted that the concepts of cloud computing and smart services are fairly new, and human-computer interaction aspects of cloud computing are scarcely defined. The personalization aspects of cloud computing may require that user-specific aspects of the cloud services need to be given attention to allow a smooth interaction between all users and their interactive interfaces.

Shneiderman [16] defines a usable interface as an interface that allows, among others, easy learnability and guidance. The challenge to provide these concepts in the cloud rather stems from the variety of the customized interfaces that can be and are required to be provided to a wide variety of user groups. For example, one user group using cloud services may consist of technology professional experts in artificial intelligence while another may consist of teenage shoppers. To provide usable interfaces that allow a smooth human-computer interaction, user requirements and demographics may need to be determined in a way that can allow for easy design adjustments.

Other usability principles from Shneiderman and Nielsen are also to some extent applicable to services in cloud computing, especially with the increase in mobile device use. A clear documentation of each mistaken done while using the services and systems may usually be difficult to provide in the cloud environment, but future service provision systems may need to pay attention to this documentation issue in design. The same attention would need to be paid to the provision of error messages in the cloud for services. Additionally, if the user intends to move away from a particular service, this should be made possible via clearly marked exits. Consistency in cloud computing is a difficult issue but cloud providers can attempt to making sure that different services are offered via consistent interfaces to allow for higher satisfaction as well as transfer of knowledge from one system to another. Service providers need to determine the specific group of users who are likely to use their services and in some cases tailor their interfaces based on the cognitive and physical needs and limitations for this group. For example, if a service caters to the elderly population, more attention should be given to the screen design in terms of contrast, colors, and font and image sizes.

One other issue where human computer interaction and usability in design may play a critical role may be administrators that use the cloud. Administrators can be at a position where they may need to access a number of different interfaces for the purposes of exchanging, sharing, manipulating and using information. To allow for a usable customer interface environment to be presented to the administrators, their job and interface requirements will need to be determined in order to determine the user requirements on their part, which can be followed by designs tailored for administrators and their specific needs on their jobs, which can vary from database to network administration and more. In the following sections, first, general personalization issues in cloud computing are discussed. This is followed by an overview of methods to gather user requirements for usability design and evaluation.

Finally, challenges in cloud computing and delivery services as they relate to usability and design are discussed.

4.1 Personalization in Cloud Computing and Smart Services

As the name indicates, smart services allow consumers to manually and easily search for and locate the services they are looking for. An analogy can be made between smart services and e-commerce retail where easy searches are done with the search box provided on e-commerce retail Web pages. Moreover, Web pages are able to give recommendations to consumers based on the shopping, browsing and searching habits of consumers. A recent study by Ozok, Fang and Norcio [23] indicated that a compact set of information with a small number of recommendations with the ability to get more information on these recommendations work best for the users. Similarly in smart services, a small number of consumer-targeted service choices with targeted service recommendations may work best in cloud computing. Personalization can be provided to users on an on-demand basis, with the interfaces offered with a default look and being able to be personalized if the service users wish to personalize them. Today, the Web, especially news-related pages, offer a number of personalization options where users can reach the relevant information and organize it according to their wishes. However, in return, some pages also ask for information on demographics, among others. Cloud service providers can use the information they gathered on their clients to tailor the interfaces to their clients' specific needs, but this should be done with permission from the clients.

4.2 Administrators, Personalization and Usability

Administrators work with the cloud by managing different applications that are most suitable for a single user or a group of users. Presenting them with a usable interface can allow administrators to work in an environment where they can maximize their performance and satisfaction with their managing and administrating duties. While service managers are software programs, a user-friendly interface can allow the controllers of such interfaces to obtain and use the relevant managing information easily from these systems. For this to happen, human factors research can allow the interfaces to be tailored to the needs of the administrators and human controllers of the manager information.

4.3 Consistency of Interfaces

Ozok and Salvendy [24] indicated that consistency plays a crucial role in computer interface design. Allowing consumers to present consistent interfaces for different applications can allow user performance and satisfaction to improve. Therefore, cloud providers can offer interface consistency for their applications to allow for consistent interfaces in the cloud. While this may be technically difficult to realize, organizations can provide at the very least a basic level of consistency to allow for smooth services and satisfied consumers.

4.4 Methods for Measuring Usability in Cloud Computing

While measuring consumer performance is almost impossible in the cloud arena, cloud users can be offered surveys to determine the usability environment of the services they are offered. Ozok [25] offered a number of usability measurement techniques using surveys. A sample representative group of cloud consumers can be given surveys to determine the critical success factors in usability as they relate to the particular cloud environment. Cloud services are variable and differ significantly between applications. However, explicit surveys that can be administered online can be used as input in usability design of the cloud environment.

4.5 Usability Challenges in Cloud Computing and Service Delivery

Cloud services widely vary in the capabilities they can offer to their consumers. A smart service can also learn from user behaviors and present user interfaces consistent with the users' cognitive capabilities, desires and satisfaction factors. Success in terms of usability in cloud computing is challenging due to the variety of services offered. However, determining the user base of the cloud, the capabilities, limitations and desires of the average cloud user can help in better interface design in cloud environment. Relevant cloud content can be presented based on user-centric approaches that start with user requirements gathering, and that involve design, evaluation and testing in terms of usability. Sample user interfaces in the cloud can also be tested in laboratory environments on small samples. While usability issues in the cloud can be expected to come more to the foreground with the advent of cloud computing and services, at this stage in development, usability concept can be seen as an important factor in smart services and on-demand application delivery.

5 Healthcare Services on the Smart Internet

The user-centric model of the Internet described so far involves the customization of websites for each user based on their preferences, functional domains, traversal patterns etc. The Internet is flooded with websites containing large volumes of data and every moment more data is being generated, distributed and made accessible all over the world. To be able to make efficient use of this data we need to apply intelligence in iterations over the data to extract information, knowledge and ultimately wisdom. The capability of customizing the website to suit individual use at the client side and extraction of useful information from data is achieved by use of smart services. They facilitate automatic detection and procurement of essential services thus proving to be beneficial to a wide variety of fields and professionals.

Health services contain extremely personalized and sensitive data and hence require a highly secure environment with high availability and accessibility. A health service system should measure for disaster recovery and fault tolerance while maintaining the high standard of data integrity. Another important aspect of executing healthcare applications is its integration with standards and compliance with Health Insurance Portability and Accountability Act (HIPAA). Some of the applications in health care that would benefit from Smart services that use Smart Internet are listed in the sections below.

5.1 Surgery

Personalized Smart service could be an extremely important tool for surgeons and physicians that would help them locate the exact article/ reference to a video or some piece of information they might want to look up urgently for the success of the surgery. While performing the surgery, a surgeon might need to refer to a similar previously recorded surgical procedure/ complication. It is extremely important for the surgeon to be able to immediately locate the required recording without having to click on each of the links returned by a text based search. The user-centric smart service used in this case can be customized to return results from medical domain. It would internally use Smart internet for an image/ video based search that would allow the surgeon to view the video of his interest at a single click. It could be further personalized based on the type of surgery or the specialization of the surgeon. For instance, in case of a cardiac surgeon performing a heart bypass surgery, the Smart service would list only videos archived under Heart Bypass Surgery category and further refine the search to result into Traditional Coronary Artery Bypass Grafting, Off-Pump Coronary Artery Bypass Grafting or Minimally Invasive Direct Coronary Artery Bypass Grafting depending on the type of bypass being currently performed on the patient. This auto-refinement of search can drastically improve the way surgeries are performed by being saving the surgeons search time in critical situations and allowing to concentrate more on the surgical complication rather than worrying about find the exact piece of information from a huge amounts of data.

5.2 Neonatal ward

One of the important goals of health care has been to reduce child mortality. According to UNICEF's records more than 70 per cent of almost 11 million child deaths every year are attributable to six causes: diarrhea, malaria, neonatal infection, pneumonia, preterm delivery, or lack of oxygen at birth. In order to help doctors to detect any life threatening infections in babies at an earlier stage, "smart" neonatal wards are used. Each heartbeat, movement of the baby in this neonatal ward is a piece of information. A Smart service in conjunction with such a neo natal ward will enable quicker knowledge gathering from the large amount of data being captured every second in the ward. It will use the information acquired from the data on Smart Internet to efficiently match the information generated in the neonatal ward and the archive of symptoms that are likely to be seen in case of life threatening infections that babies are vulnerable to. Thus it can analyze the data and predict what can happen, faster and help doctors take action earlier. Thus smart services can be effectively used in preventive cure of diseases by diagnosing symptoms and suggesting appropriate preventive treatment to avoid the disease.

5.3 Medical Record Synchronization

Traditionally medical records have been written on paper and kept in folders. The advent of electronic medical records has not only changed the format of medical records but has also increased accessibility of files. In case of an emergency situation where a patient is brought into a hospital that doesn't hold his previous medical record, acquiring this information can be a time-consuming job. In cases where the patient is allergic to a particular medicine, it is very important to have his previous medical record handy before being administered any drug. Smart services can play a big role in this situation. It can allow a service provider to maintain a person's medical record right from birth along with links to family's medical history. This record gets augmented with more information with his every visit to a doctor. Such a record will be a one-stop medical database for that patient. When the person is hospitalized, hospital with legal authority can use the Smart service to access his medical records. Thus doctors can immediately get an entire picture of the patient's medical history, allergic conditions, medications being taken etc. Moreover, the smart service will also use the patient's family member's medical records and suggest the doctor of a potential risk the patient might face due to heredity even though it doesn't show in his records. This can save the doctor's valuable time by being able to link trees of data, helping them diagnose the potential problems and decide on the safest treatment thus ensuring higher chances of success in saving lives. The challenge faced in developing such an application would be having centralized patient data available for use at any time.

5.4 Telesurgery

Telesurgery (also known as Remote surgery) allows a surgeon to perform surgery on a patient who may not be physically present in the same location. It combines elements of robotics, communication technology such as high-speed data connections and management information systems. Smart services coupled with the telesurgical equipment could sense the current state of the surgical procedure, lookup the medical records of the patient, search similar surgeries previously performed and suggest the surgeon of alternative steps he could follow in the surgery. This is analogous to having another "experienced" surgeon in the room to assist the operating surgeon with the right knowledge of the surgery and the patient. An example of this would be a cardio surgeon looking up a video of previously performed bypass surgery for reference, on the internet, while performing a remote surgery. The accurate result of the lookup must be obtained very quickly considering the time criticality of the current surgery. The smart services pre-customized to the surgeon's search pattern and requirements would eliminate a large amount of time required to return the desired video. The smart service would not just return the correct reference but also provide suggestions for controlling the telesurgical equipment under gives circumstance. Thus a smart service can be customized to every user's needs and specifications allowing the user to get desired results. The challenge faced in such an application would be the development of an efficient filtering mechanism that allows the smart service to return the most accurate and relevant results and a control mechanism that coordinates with the surgical equipments.

5.5 Cloud Based Medical Image Visualization

Cloud based medical image visualization is a unique concept of centralizing and sharing all radiology images such as CT-Scans, MRIs, X-rays, ultrasound among others in a secured environment. Such cloud based environment can seamlessly integrate all medical imaging with electronic patient records that can be accessed from any location and reviewed by any authorized user. In such scenario, it is extremely important to store and transmit medical records in a secured and safe environment with a very high standard of data integrity, protecting patient privacy and complying with all security and HIPAA regulations.

Medical images are large files (Digital Imaging and Communication in Medicine (DICOM) objects) and therefore, it is difficult to share among several care delivery organizations. There is significant amount of delay involved in sharing these medical images. Such delays often result in repetition of medical procedures and increased cost of healthcare to the patients, insurance companies and federal government. In addition to the cost saving, sharing of medical records, specifically radiology imaging databases, can drastically reduce medical redundancies and exposure to radiations. A DICOM object viewer and the Picture Archiving and Communication System (PACS) server can reside in a cloud computing environment.

A key requirement for DICOM viewers is lossless image coding; users accessing DICOM images should receive lossless image to rule out any compression artifacts. The views rendered by the DICOM viewer have to be communicated to the users remotely accessing the image. Commercial remote access tools such as Citrix use lossy compression for remote viewing and hence are not suitable for medical imaging application. The use of lossy compression may not be an acceptable solution under several medical conditions. For instance, a lossy compression may provide wrong information about the size of a cancer cells that may be growing in any part of a body. Since the stage of a cancer is determined by the volume of the cancer cells; a lossy image may show a reduced volume by removing some pixels.

A lossless virtual presentation layer can then be developed to reside on cloud. This component can allow radiologist to annotate the image through a web based viewer and store it back into the distributed cloud based database. Therefore, an instantaneous lossless access to all DICOM objects can be achieved to eliminate the download time.

6 Multimedia Services on the Smart Internet

In a user centric model for content access on the Internet, content is adapted and customized for the user needs. The multimedia nature of the content makes the adaptation problem complex – computationally as well as algorithmically complex. Customizing a website to suit individual use requires user context awareness and the customization possible depends on type of content used on the website. Customizing content such as video is highly resource intensive. We showed that customization based on computational model of human attention is a bandwidth conserving solution but requires large computational resources [26]. This section presents challenges and possible approaches to overcome the challenges in offering video services over cloud infrastructure. The Smart Internet would adapt one of these approaches.

Multimedia content accessed by the users is either stored or real-time. Stored media is pre-recorded and hence can be pre-processed to enable rapid customization when users access such content. Real-time media, on the other hand, is generated in real-time and requires real-time computing resources to perform the desired customization. In the case of video and audio, customization typically implies changes to one or more of bit rate, resolution, quality, and modality [27]. Customization could also involve knowledge extraction which in turn will help in customizing related content and services.

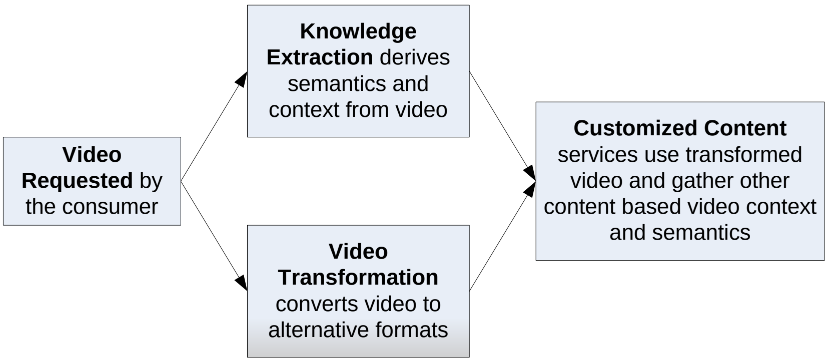

Figure 2: User centric customization of video content

Video is data intensive media with large computational requirements for processing and can potentially benefit from cloud computing infrastructure. The parallelism inherent to video processing problems depends on the processing at hand and the format of the video. Since video is necessarily compressed for transmission and storage, any video processed by the cloud is typically compressed. Video compression, while reducing storage and bandwidth requirements, creates dependencies that could affect the granularity of parallelism possible in a given video processing task. For example, predictive coding modes in video use previously coded frames to improve compression efficiency but create dependencies that are undesirable in distributed computing. A key challenge is balancing the tradeoffs between compression efficiency and the possibilities for parallelism.

Smart Internet infrastructure would be able to customize user centric video services by rapidly transforming videos to suite the current user context and also use the knowledge from videos to create a better experience for users. For customizing stored video, the challenge is to reduce the turnaround time. Typically turnaround time can be reduced by increasing the computing resources – e.g., number of machine instances on the cloud dedicated to the given task. A map-reduce framework such as Hadoop can thus be used to reduce the turnaround time. In the case of video, the number of machine instances cannot be increased arbitrarily and is limited by the number of video segments that can be independently processed. The performance and scalability of a video processing problem is thus influenced by the video compression features used (e.g. number of I frames as I frames typically create independent video segments). This also implies that the framework for workload distribution on the cloud has to be media aware.

Real-time video is generated by a single source and improving the throughput, i.e., increasing the number of streams processed, is a key performance consideration for a service provider. Depending on the type of service, real-time video services allow a few seconds of delay and the buffered video can be used to parallelize the problem and increase the throughput. The real-time nature of the video data stream puts constraints on how the performance can be improved. Solutions such as load balancing are more appropriate and distributed computing frameworks such as Hadoop are not suitable.

7 Conclusion and ongoing work

User-centric smart services will create a paradigm shift in the way management perceives IT in the organization. In this chapter we have detailed our vision for services lifecycle in the Smart Internet. We have described the service manager, monitor, procurer and integrator applications that will need to be developed. To the best of our knowledge, this is the first such effort, and it provides a description of the new applications needed to automate discovery, acquisition and deployment of services. As part of our ongoing work, we are developing policies for service procurer using the Semantic Web technologies. Usability issues in cloud computing also need to be explored further to provide environments for cloud service users that are more effective, efficient and user friendly.

References

1. M Xu, Z Hu, W Long, W Liu, Service virtualization: Infrastructure and applications - The Grid: Blueprint for a New Computing Infrastructure by Ian Foster, Carl Kesselman, Morgan Kaufman, (2004)
2. J. Ng, M. Chignell, J. Cordy, The Smart Internet Transforming the Web for the User, Sep 2009, Manuscript under preparation
3. M. Papazoglou and W. Van Den Heuvel, Service-oriented design and development methodology, International Journal of Web Engineering and Technology, vol. 2, Number 4, pp. 412 – 442 (2006)
4. D. Bianchini, V. De Antonellis, B. Pernici, P. Plebani, Ontology-based methodology for e-service discovery, International Journal of Information Systems, The Semantic Web and Web Services, Volume 31, Issues 4-5, June-July 2006, pp 361-380 (2006)

5. L Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, Q. Sheng, Quality driven web services composition, Proceedings of the 12th international conference on World Wide Web, pp 411 - 421 (2003)
6. E. M. Maximilien, M.Singh, A Framework and Ontology for Dynamic Web Services Selection, IEEE Internet Computing, vol. 8, no. 5, pp. 84-93, Sep./Oct. (2004)
7. J. Black et al, An integration model for organizing IT service Management, IBM Systems Journal, vol. 46, no 3, (2007)
8. Web Services Description Language (WSDL) 1.1, http://www.w3.org/TR/wsdl, March (2001)
9. D. McGuinness, F. Van Harmelen, et al., OWL web ontology language overview, W3C recommendation, World Wide Web Consortium, (2004).
10. O. Lassila, R. Swick and others, Resource Description Framework (RDF) Model and Syntax Specification, World Wide Web Consortium, (1999)
11. Web services Business Execution Language (WS-BPEL) 2.0, http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf
12. L. Kagal, T. Finin, A. Joshi, A Policy Based Approach to Security for the Semantic Web, In proceedings of 2nd International Semantic Web Conference (ISWC2003), September (2003)
13. Anupam Joshi, Z. Jiang, Retriever: Improving Web Search Engine Results Using Clustering, in Managing Business with Electronic Commerce: Issues and Trends, (A. Gangopadhyay, Editor), Idea Press, (2001)
14. W.T. Balke, M. Wagner, Towards personalized selection of web services, (WWW 2003) Alternate Track on Web Services, (2003)
15. Stanford Personalized PageRank Project, http://nlp.stanford.edu/projects/pagerank.shtml, last retrieved on October 12 2009
16. Shneiderman, B., Designing the User Interface: Strategies for Effective Human-Computer Interaction. New York: Addison-Wesley, 1992
17. J. Nielsen, Usability Engineering, Morgan Kaufmann Publishers Inc., San Francisco, CA, 1993
18. S Ran, A model for web services discovery with QoS, ACM SIGecom Exchanges, Vol 4, Issue 1, 2003, pp 1-10, (2003)
19. Microsoft System Center: Service Manager, http://www.microsoft.com/systemcenter/en/us/service-manager.aspx, retrieved April 5 2010
20. IBM Service Management, http://www-01.ibm.com/software/tivoli/solutions/, retrieved April 5 2010
21. Oracle Web Services Manager, http://www.oracle.com/appserver/web-services-manager.html, retrieved April 5 2010
22. Hewlett-Packard Service Manager software, retrieved April 5 2010, https://h10078.www1.hp.com/cda/hpms/display/main/hpms_content.jsp?zn=bto&cp=1-11-85^12473_4000_100__&jumpid=reg_R1002_USEN
23. Ozok, A. A. Fan, Q. and Norcio, A., Design Guidelines for Effective Recommender System Interfaces Based on a Usability Criteria Conceptual Model. Behaviour and Information Technology, Vol. 29(1), pp. 57-83.
24. Ozok, A. A. and Salvendy, G., How Consistent is Your Web Design? Behaviour and Information Technology, Vol. 20(6), pp. 433-447.
25. Ozok, A. A., Survey Design and Implementation in Human Computer Interaction. The Human-Computer Interaction Handbook Second Edition by Jacko, J. and Sears, A. (Eds.), Lawrence Erlbaum Associates, Mahwah, NJ, pp. 1151-1169.
26. R. Jillani and H. Kalva, “Exploiting spatio-temporal characteristics of human vision for mobile video applications,” (Invited Paper) Proceedings of SPIE Volume 7073, Applications of Digital Image Processing XXXI, part of SPIE Optics + Photonics 2008, San Diego, CA.

27. A. Vetro and H. Kalva, “Technologies and Standards for Universal Multimedia Access,” Handbook of Video Databases, B. Furht, Ed., ISBN 0-8493-7006-X, CRC Press, May 2003.
28. Web Services Policy Framework, Contributed by IBM, BEA Systems, Microsoft, SAP AG, Sonic Software, VeriSign, (http://www.ibm.com/developerworks/library/specification/ws-polfram/), retrieved on April 26 2010.
29. Dustdar, S., and Schreiner, W., A Survey on web services composition, Int. Journal on Web and Grid Services, InderScience, vol. 1(1), pp. 1-30, 2005.
30. Milanovic, N and Malek, M, Current solutions for Web service composition, IEEE Internet Computing. Vol. 8, no. 6, pp. 51-59. Nov.-Dec. 2004.