Enterprise Cloud Service Architecture

Longji Tang, Jing Dong, Yajing Zhao
Department of Computer Science
University of Texas at Dallas
Richardson, TX 75083, USA
jdong@ieee.org

Liang-Jie Zhang
IBM T.J. Watson Research Center
19 Skyline Drive
Hawthorne, NY 10532, USA
zhanglj@ieee.org

Abstract

Cloud computing, a new paradigm of distributed computing, introduces many new ideas, concepts, principals, technologies and architectural styles into enterprise service-oriented computing. The enterprise service-oriented architecture (ESOA) style is an abstraction of concrete enterprise service-orientated architectures, which includes SOA architectural elements, service design patterns as well as principles, and SOA quality attributes. It can be extended to a new style for realizing enterprise cloud computing. Meanwhile, the principles and style of enterprise service-oriented computing facilitate the enterprise-wide adoption of cloud computing. This paper extends the ESOA style to a new hybrid architectural style, Enterprise Cloud Service Architecture (ECSA). The style is described by extending enterprise service-oriented formula for ESOA. We model the style through specifying each element in the formula with both service-oriented and cloud architectural styles.

1. Introduction

With the globalization of economic and environment, increasing complexity of business processes makes the enterprise information systems more and more complicated. Enterprise service-oriented architecture (ESOA) is designed to tackle the complexity and build better architectures and solutions for enterprise[1]. ESOA is a hybrid architectural style of SOA style and Enterprise Architectural (EA) style, which defines the concrete ESOA architecture as a set of well-defined services. It may be further abstracted to process layers and composite applications for business solution. The services are deployed and accessed through SOA infrastructure. They are governed and managed by SOA principles and management systems [11][12][27][30][32][34]. ESOA brings the agility aspect to enterprise architecture, allowing enterprise to deal with system changes using a configuration mediation layer, rather than constantly redeveloping these systems. However, ESOA introduces new challenges and issues to enterprise architecture due to its on-premise characteristics:

- Enterprise owns data center with ESOA services and the infrastructure is not dynamic such that it does not support auto scaling and elastic load balancing [2];
- Enterprise architecture is built behind firewalls;
- Resources are dedicated to each workload;
- Resources are shared within enterprise only.

Figure 1 shows a traditional ESOA data center with three-layer infrastructures:

- Web server infrastructure,
- Enterprise application server and service infrastructure which includes application database and SOA services, application monitors and SOA application management,
- Enterprise information storage and business service infrastructure.

All enterprise services are running behind firewalls.

Building a data center to support ESOA architecture is expensive. It is impossible for some small to medium enterprises. For large enterprise, it is difficult to complete some complicated business processes, such as online shopping and shipping, without third party services. Moreover, many server resources in a large data center are idle or passive, such as in non-peak time, since the plan of resources is based on the highest volumes of workload. Thus, resources are wasted resulting in increasing cost of resources and operations. Many enterprises view SOA as something that is only applicable within firewall. The ESOA is facing new challenges from enterprises – reducing complexity as well as cost, and increasing capacity, flexibility as well as agility. Cloud computing, a new paradigm of distributed computing applied to enterprises, brings many new ideas, concepts, solutions, principles to enterprise architecture and ESOA. Originally, cloud computing is a convergence of Internet computing (such as web 2.0 [15]), service-oriented computing [32][30][12], grid computing [33], utility computing [7], virtualization [18], and virtual applications. Cloud computing is about sharing services, computation, and/or data off-site via an internal or

[1] This paper considers the terms, enterprise service-oriented architecture (ESOA) and service-oriented enterprise architecture (SOEA), are interchangeable.

978-0-7695-4130-3/10 $26.00 © 2010 IEEE
DOI 10.1109/CLOUD.2010.10

external, location-transparent, centralized facility or contractor for lower cost and business benefits. Services and data, made available in the cloud, can be more easily and ubiquitously accessed, often at much lower cost, increase their value by creating opportunities for enhanced collaboration, integration, and analysis on a shared common platform [9]. Therefore, adding cloud computing to ESOA takes it to the next level and expands it from on-premise to off-premise.

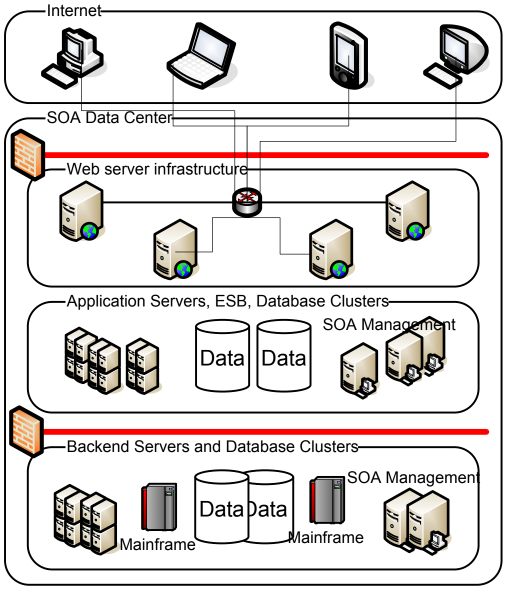

Figure 1 Enterprise SOA Data Center

We have proposed a *service-oriented* formula, defined in terms of EWS-* substyle, for modeling ESOA styles in [27][29]. The formula is used to specify another ESOA substyle, EWOA, in [28]. This paper extends the formula and specifies a new architectural style, Enterprise Cloud Service Architecture (ECSA), which is hybrid of ESOA style and Cloud Computing (CC) architectural style. The motivations of our research work on the ECSA include specifying, understanding and evaluating ECSA architectural style through common principles and constraints of a family of ECSA enterprise architectures.

The remainder of this paper is organized as follows: the next section defines the new style ECSA by our extended formula. The last two sections cover related work and conclusions.

2. Enterprise Cloud Service Architecture

This section defines the new ECSA style by extending the enterprise *service-oriented* formula [29]. Before defining the formula, we define

$$Cloud = \{\text{public cloud}\} \cup \{\text{private cloud}\} \quad (2.1)$$

in which the public cloud, such as Google cloud [37] and Amazon cloud (Amazon Web Services - EC2 and S3) [25], is typically on the Internet or off-premise. The private cloud, such as cloud-enabled data center, is typically located on-premise.

The architectural style ECSA can be modeled by the following 8-tuple:

$$ECSA = \langle S, C, D, SI, SM, SP, SQ, SD \rangle \quad (2.2)$$

where

$$S = S^{I} \cup S^{II} \cup S^{III} \quad (2.3)$$

in which

$S^{I} = \{s \mid s \text{ is a service not on cloud}\}$

$S^{II} = \{s \mid s \text{ is a service on private cloud}\}$

$S^{III} = \{s \mid s \text{ is a service on public cloud}\}$

$S^{IV} = \{s \mid s \text{ is a service on hybrid cloud}\}$

$S^{V} = \{s \mid s \text{ is a service on community cloud}\}$

where $S^{IV} \subseteq S^{II} \cap S^{III}$ and $S^{V} \subset S^{III}$.

$$C = C^{I} \cup C^{II}, \quad (2.4)$$

in which

$C^{I} = \{c \mid c \text{ is a non-cloud service consumer}\}$

$C^{II} = \{c \mid c \text{ is a cloud service consumer}\}$

$$D = D^{I} \cup D^{II}, \quad (2.5)$$

in which

$D^{I} = \{d \mid d \text{ is a SOA data element}\}$

$D^{II} = \{d \mid d \text{ is a cloud data element}\}$

$$SI = SI^{I} \cup SI^{II}, \quad (2.6)$$

in which

$SI^{I} = \{r \mid r \text{ is a SOA infrastructure}\}$

$SI^{II} = \{r \mid r \text{ is a cloud infrastructure}\}$

$$SM = SM^{I} \cup SM^{II}, \quad (2.7)$$

in which

$SM^{I} = \{m \mid m \text{ is a SOA management}\}$

$SM^{II} = \{m \mid m \text{ is a cloud management}\}$

$$SP = SP^{I} \cup SP^{II}, \quad (2.8)$$

in which

$SP^{I} = \{p \mid p \text{ is a SOA process}\}$

$SP^{II} = \{p \mid p \text{ is a cloud process}\}$

$$SQ = SQ^{I} \cup SQ^{II}, \quad (2.9)$$

in which

$SQ^{I} = \{q \mid q \text{ is a SOA quality attribute}\}$

$SQ^{II} = \{q \mid q \text{ is a cloud quality attribute}\}$

$$SD = SD^{I} \cup SD^{II} \cup SD^{III}, \quad (2.10)$$

in which

$SD^{I} = \{d \mid d \text{ is a building element of development}\}$

$SD^{II} = \{d \mid d \text{ is a service deployment type}\}$

$SD^{III} = \{d \mid d \text{ is a service delivery model}\}$

Formula (2.2) extends the enterprise *service-oriented* formula from the ESOA style to the new ECSA style. From the definitions of each element of (2.2), the ECSA combines both ESOA style and cloud computing style. In the following subsections, we will describe each element defined in formula (2.3) through (2.10) in details.

2.1 A 3D Model of Cloud Services

Compared with the enterprise *service-oriented* formula, only one new element *SD* is added into the ECSA formula (2.2). We call *SD* the 3D model of Cloud Services. The 3D model distinguishes between traditional ESOA services and cloud services as well as between ESOA style and ECSA style. Cloud computing allows enterprises to share computing resource globally through the Internet [31]. The resource sharing results in the 3D model. In the 3D model, SD^{I} is a set of building blocks and tools for cloud services and enterprise applications development, so it is the base of developing the services in {PaaS} = {Platform as a Service}. SD^{II} is a set of deployment types of cloud services as follows.

SD^{II} = {PrC, PuC, VPC, CoC, HyC}

Deployment Type	Description
PrC	Private Cloud [21]
PuC	Public Cloud [21]
VPC	Virtual Private Cloud [36]
CoC	Community Cloud [21]
HyC	Hybrid Cloud [21]

Table 1 Deployment Type of Cloud Services

Table 1 provides the description for each type. The CoC can be managed by the community or third party and exist on-promise or off-promise. The VPC is first created by Amazon [36]. It is a private cloud lives in a public cloud through the VPN network. The customer's resources are isolated in the public cloud, which provide an online virtual data center to customers. The CoC infrastructure is shared by a community – several enterprises or organizations sharing the same concerns, such as mission, security and policy requirements.

SD^{III} is a set of delivery modes of cloud services as follows:

SD^{III} ={SaaS, PaaS, IaaS, IMaaS, IRaaS, XaaS}

Table 2 provides the description of each mode:

Delivery Mode	Description	Resource sharing
SaaS	Software as a Service [25]	Sharing software
PaaS	Platform as a Service [21]	Sharing platform
IaaS	Infrastructure as a Service [21]	Sharing infrastructure
IMaaS	Information as a Service [21]	Sharing information
IRaaS	Integration as a Service [21]	Sharing integration
XaaS	Other cloud service delivery model [21]	Sharing other resources

Table 2 Delivery Modes of Cloud Services

2.2 Cloud Services

We have formally and informally specified services as a self-contained software abstraction of business, technical functionality, or infrastructure management, defined by a well-defined interface [11]. We define the kind of enterprise services as functional services which serve business for completing certain operations, such as shopping transaction web service and hotel reservation web service. They include composed and process services, such as workflow services. If a functional service s is not exposed to the internet (out of enterprise firewall) or it cannot be accessed from the internet, then $s \in S^{I}$. In this paper, we focus on the managed services (or enterprise services) on the cloud. We define an Enterprise Cloud Service (*ECS*) as a specific managed service with Service Level Agreement (SLA), elasticity/dynamism, accountability/utility, loose-coupled, which can be accessed and delivered through the internet.

If s is an *ECS* within the enterprise internal network, then $s \in S^{II}$ is called private cloud service.

If s is an *ECS* in an enterprise cloud service provider network, then $s \in S^{III}$ and s is called public cloud service.

Cloud computing extends the ESOA service concept and capacity to broader area in two aspects – the vertical and horizontal aspects as shown in Figure 2.

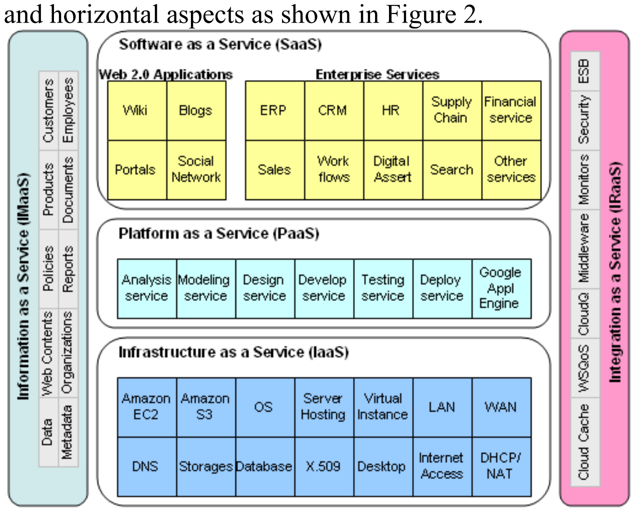

Figure 2 View of Cloud Services

We define

$$S_{ESOA} = \{s \mid s \text{ is a traditional SOA service}\}$$

and

$$S_{Cloud} = \{s \mid s \text{ is a } \textit{Cloud} \text{ service}\}$$
$$= S^{I}_{Cloud} \cup S^{II}_{Cloud}$$

in which S^{I}_{Cloud} = {SaaS} ∪ {PaaS} ∪ {IaaS}, i.e., S^{I}_{Cloud} includes three kinds of basic cloud services; and S^{II}_{Cloud} = {IMaaS} ∪ {IRaaS} ∪ {XaaS}, i.e., S^{II}_{Cloud} includes the rest types of cloud services. Thus, $S_{ESOA} \cap S_{Cloud} \neq \emptyset$. If a service $s \in S_{ESOA} \cap S_{Cloud}$, then $s \in S^{II}$.

To specify an *ECS*, let us define the following sets of properties:

- Cloud Service Interface Type

  I_{type} ={User Interaction Interface, Web Service Interface, REST Interface, Web Application Interface, Event Interfaces}

- Cloud Service Access Type
  A_{type}={a | a is a client access protocol method}, such as Web User Interaction (HTTP), Web Service API (SOAP), REST API (HTTP), Web Application API, Event Trigger, distributed devices (wireless devices).
- Cloud Service Provisioning Type
  P_{type}={Applications, Business Operations, Resources, Information, Platform, Integration}
- Cloud Service Control/Ownership Type
  $O_{type}=\{O_{own}, O_{thirdparty}\}$, in which O_{own} = Buy/lease and Own, $O_{thirdparty}$ = Owned by public cloud provider and pay-as-you-go

Now, we can specify an *ECS* as an 8-tuple:

$$ECS = \langle I_{ECS}, A_{ECS}, P_{ECS}, O_{ECS}, SD^{II}_{ECS}, S^{III}_{ECS}, Policy_{ECS}, SQ_{ECS} \rangle$$

where $I_{ECS} \subset I_{type}$, $A_{ECS} \subset A_{type}$, $P_{ECS} \in P_{type}$, $O_{ECS} \in O_{type}$, $SD^{II}_{ECS} \in SD^{II}$, $SD^{III}_{ECS} \in SD^{III}$, $SQ_{ECS} = QoS_{ECS} + SLA_{ECS} \subset SQ$ in which QoS_{ECS} is the *ECS* Quality of Service and SLA_{ECS} is the *ECS* Service Level Agreement between *ECS* provider and its consumer. For instance the Amazon EC2 cloud service [25] can be specified as $EC2_{Amazon}$ = $\langle I_{EC2}, A_{EC2}, P_{EC2}, O_{EC2}, SD^{II}_{EC2}, S^{III}_{EC2}, Policy_{EC2}, SQ_{EC2} \rangle$ where I_{EC2} = {Web service interface, REST interface}, A_{EC2}={Web service API, REST API}, P_{EC2}= resources, O_{EC2} = $O_{thirdparty}$ (owned by Amazon), SD^{II}_{EC2} =PuC, SD^{III}_{EC2} =IaaS, ($Policy_{EC2}$, SQ_{EC2}) has two parts – the documentation which can be found from [1] and runtime policy and SLA which are managed by Amazon's runtime cloud management.

2.3 Cloud Service Consumers

We have specified the ESOA service consumers C^{I} and part of the consumers of private cloud services in (2.4) [27][29]. The part of ESOA service consumers are also part of consumers of private cloud services. In this paper, we focus on specifying the enterprise cloud service consumers C^{II}. In Figure 3, we show that there are four kinds of enterprises with different ECSA architectural styles:

- Enterprise A has no data center and it is a consumer of public cloud of the provider Enterprise B. Most small to medium enterprises typically are or will become this kind of enterprise.
- Enterprise B is a public cloud provider which provides public cloud services, such as Amazon cloud [1], Google cloud [37], Saleforce cloud [25], IBM cloud center [18], and Microsoft Azure cloud [22].
- Enterprise C has data center with private cloud services whose consumers are cloud applications accessed by internal customers, such as registered users, employees and partners. The private cloud services can be the consumers of other public cloud services in SEDC (somebody else's data center).
- Enterprise D has multiple data centers and hybrid clouds. The consumers of its public cloud services can be private cloud inside the enterprise, and internal and external cloud applications assessed by external clients that include external end-users and cloud applications in other enterprises. The consumers of its private cloud services can be internal applications accessed by internal clients that include internal end-users and the public cloud services within the collocation. Most large enterprises are or will become this kind of enterprise.
- The cloud service consumers also depend on the type of the cloud service. If the *ECS* is in {PaaS}, such as Google App Engine, then web software developers, IT managers and application system administrators are the consumers of *ECS*. If the *ECS* is in {IaaS}, then the system and database administrators are its consumers.

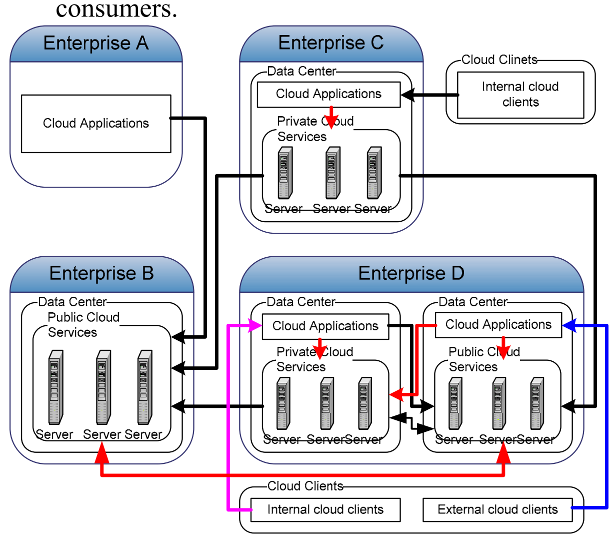

Figure 3 Cloud Service Consumers

Specifically, the public or private cloud service consumers have the following characteristics:

- Self-service: users can access services they provide or directly procure services in the cloud. Users also manage and monitor cloud services from self-service portal.
- Standard API for accessing cloud services
- Rapid service provisioning
- Pay-for-use

2.4 Cloud Data

The set D in (2.5) consists of two sets of ECSA data elements which are used for building ECSA style enterprise architecture (EA). The SOA data set D^I has been specified in [27][29]. Part of the data and metadata in D^I are also used by cloud services, infrastructure and management, such as various resources and their profiles, basic infrastructure configuration data, and SOA metadata. However, cloud computing needs some cloud specific data and metadata in D^{II} as shown in Table 3.

Cloud Data	Examples from public or private cloud
Virtual resources	Virtual instance, virtual server, virtual OS, virtual network, virtual storage
Application metadata	Google App Engine application metadata
Cloud policy	Security policy, Routing policy, Privacy policy, Access policy (such as Amazon web services REST/SOAP access control policy)
Cloud SLA	Error rate, Monthly update percentage, Service credit, Region Unavailable
Utility model data	Pricing (such as EC2 high CPU on-demand instances – Medium UNIX $0.20/per hour), Billing, Paying for what user used.
Virtualization metadata	The virtualization metadata contains all setup and configuration information required for the virtualization layer to establish a connection and it may also contain additional information to make some specific operations (examples of metadata are: server name, database name, user, password, translation fields, etc.). It is usually described by a XML schema and stored in metadata repositories or database.
Application network delivery metadata	It includes all setup and configuration information required for application delivery infrastructure, such as load balancing, acceleration, optimization, and security.
Infrastructure instance metadata	EC2 instance metadata [3]
Cloud configuration data and metadata	Types of resources (such as CPU, Storage, OS, Software, Monitoring), Types of instances (such as Amazon EC2 – Small, large and extra large instances, High-CPU medium and extra large instances)

Table 3 Cloud Data

2.5 Cloud Infrastructure

The traditional SOA infrastructure SI^I is the heart of ESOA. It is the bridge of the transformation between business and services. For the new ECSA style, the cloud infrastructure is added. It is easy to show $SI^I \cap SI^{II} \neq Ø$ which means the infrastructure of the new style is hybrid of both SOA and cloud infrastructure styles. The traditional ESOA infrastructure is not dynamic and flexible enough. Therefore it is not adaptable to today's on-demand business workload and real-time B2B requirements. It also costs more resources and power in enterprise's data center. The cloud infrastructure SI^{II} is a dynamic IT infrastructure which consists of elastic web servers, elastic application servers, elastic MQ servers and elastic database servers. It has the following main characteristics:

- It supports elasticity and dynamism – automatic scalability and load-balancing, failover in terms of virtualization [7][9][16] or other technologies [37].
- It supports global resource sharing through the Internet.
- It supports resource usage accountability – utility model [7][31].
- It can be a part of cloud service, such as PaaS type services (Google App Engine [37]), or can be a cloud service, such as IaaS type service (Amazon EC2).

Therefore the cloud infrastructure brings cost-effective ways and elasticity to current SOA infrastructure. The SOA infrastructure is now moving to a dynamic cloud-enabled infrastructure as shown in Figure 4.

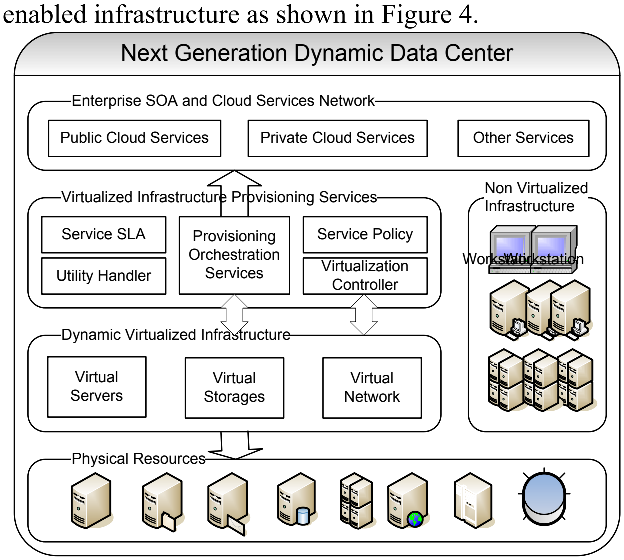

Figure 4 Dynamic Data Center

2.6 Cloud Management

Cloud computing is changing the landscape of ESOA and bringing new type of services and dynamic infrastructure into ESOA. An enterprise architecture needs SOA to achieve better quality and to leverage cloud computing providers [21]. The relatively mature SOA management or governance SM^I is the base of cloud management SM^{II}. It is easy to show that $SM^I \cap SM^{II} \neq$ Ø. The SOA management we have specified in [27][29], such as network and application monitoring, identity management, policy enforcement, service-level agreement management, and service lifecycle management in SM^I, is very important for cloud computing. Thus they are also in SM^{II}. However, cloud computing brings the SOA management to a new level from two aspects: enhancing SOA managements and adding some new cloud specific managements, since

- Cloud requests more dynamic, real-time and automatic runtime governance to services and elastic infrastructure.
- Cloud requires automatic policy and SLA management at runtime
- Cloud requests an automatic service provisioning and subscription management for its utility model. There are two new cloud services – cloud provisioning service and cloud subscription service which are used for managing cloud provisioning and subscription [31].
- Cloud requests new identity management for cloud service security and trust challenges [6], such as Amazon cloud security process [4].

2.7 Cloud Processes

One of important parts of ESOA style is its set of SOA processes. The SOA process or workflow is an abstraction of Business Process Management (BPM). Each process is composed by multiple services in orchestration and/or choreography for completing a whole or partial business process or task. The traditional SOA process can be executed by using an ESOA infrastructure with process engine in the internal network of an enterprise. However the traditional SOA processes face many challenges and issues: real-time high performance (such as automated trading), on-demand scalability, and large payloads (10+ MB), memory constraints, and high availability and reliability. In a distributed SOA environment of an enterprise, the bottlenecks tend to occur in all, two or one of the following three places [8]:

- Shared intermediary services,
- The services themselves, and
- SOA infrastructure operations.

The scalability bottlenecks across all these SOA parts in workflow/process are caused when disk I/O, memory, or CPU saturation levels are reached in most cases. Moreover the cluster technology, adopted by traditional SOA, can provide higher availability. However it depends on static partitioning, where a single backup server is pre-assigned to service requests from a failing server. The grid-enabled SOA [8] provides a way to improve the performance, scalability and availability of SOA processes. Cloud computing shares the same goal as grid computing, that is to allow service consumers to obtain computing resources on-demand. However, cloud computing is a new style of distributed computing, which introduces many new architectural styles and technologies to SOA. There are four aspects that cloud computing differ from grid computing [14]:

- It is massively scalable.
- It can be encapsulated as an abstract entity that delivers different levels of services to the customers outside of the Cloud.
- It is driven by economies of scale.
- The services can be dynamically configured through virtualization or other methods and delivered on-demand.

Cloud-enabled SOA process [5] not only provides new BPM delivery model, but also improves SOA process's scalability, performance and availability in terms of cloud computing.

2.8 Cloud Quality Attributes

The software architectural quality attributes [24] include not only the principles of system architecture design, but also the non-functional constraints of structure and behavior of any software architecture. Therefore we include the architectural quality attributes as part of ECSA. We have defined common SOA quality attributes SQ^{I} of ESOA style in [27][29]. They are also quality attributes of cloud computing, specifically, private cloud. Therefore, $SQ^{I} \cap SQ^{II} \neq \emptyset$. They both share many commonalities, such as performance, security, scalability, availability, however,

(1) the quality attributes of SOA and public cloud have different degrees of maturity. In general, the maturity of cloud quality attributes is less than that of SOA quality attributes;
(2) the specifications of some of cloud quality attributes are different from traditional ESOA, such as elastic scalability; and
(3) SQ^{II} includes some cloud-specific quality attributes and properties of cloud services, such as cloud visibility and subscription.

3. Related Work

Current work on bridging ESOA and Cloud Computing (CC) can be classified in the following categories, (1) specifying and analyzing cloud service-oriented architectural style or framework, (2) CC and SOA convergence in enterprises, and (3) creating new approaches combining SOA approaches and cloud computing which include bringing SOA best practice into cloud computing and adopting cloud computing power for improving existing ESOA architectures.

Zhang et al. [31] propose a Cloud Computing Open Architecture (CCOA). The CCOA is a cloud computing centric service-oriented architecture framework which bridges the power of SOA and virtualization in the context of Cloud Computing ecosystem. Seven principles of cloud computing architecture are also presented in

[31]. This paper proposed a new hybrid architecture style specified by a cloud service-oriented formula (2.2) which includes basic architectural elements, service orientation and cloud principles.

Many SOA software venders, like IBM [18], HP [17], SUN [26], Oracle [23], and Microsoft [22], propose their new software product architectural models and frameworks which combine the SOA and cloud computing powers. The model presented in this paper can be used for evaluating different architecture approaches.

Linthicum presents the dream team of cloud computing and SOA in [21]. He points that SOA and cloud computing provide a great deal of value when they work together. He describes the relationship of SOA and cloud computing and guides enterprises on how to make cloud computing and SOA convergence step-by-step. We describe the relationship of SOA and cloud computing through the hybrid architectural style.

Lakshman [19] shows how cloud stretches the SOA scope and proposes a process for identifying cloud scenario. Its case studies show Microsoft cloud Azure integrates to enterprise SOA system.

Lawson [20] points that SOA can help cloud computing at three aspects: (1) one can move services around as needed, including to a cloud server, to address pressing business needs; (2) One can take advantage of virtualization or, as Linthicum explains, address “core applications as logical instances that may run on any number of physical server instances, providing better resource utilization, and scalability”; (3) One can create mashups or on-the-fly composite apps with services and tap the cloud's computing power.

Varia [35] introduces the cloud architectures of Amazon Web Services (AWS). AWS is a typical example of adopting ESOA’s web services (SOAP or REST) to their cloud architectures. DeCandia et al. [10] presents Amazon service-oriented cloud architecture through Dynamo design and how SOA governance can help cloud in achieving high performance and availability.

Dornemann [13] proposes an approach that extends an open source SOA BPEL implementation to use Amazon’s EC2 for providing the process dynamic resources.

Many literatures, such as [25][35][37], describe Amazon cloud, Google cloud, Saleforce cloud. Their architectures can be shown as the instances of the hybrid style ECSA.

Compared with all other approaches to modeling enterprise cloud service-oriented architecture, our approach differentiates other approaches in the following aspects:

- We define ECSA as a new hybrid architectural style with ESOA style and cloud computing style.
- The ECSA is specified based on our tuple formula (2.2) which is an abstraction of the family of enterprise architectures building on ESOA and Cloud computing.
- The ECSA as a top style is extensible. It can have several substyles – Private ECSA style, Public ECSA style and hybrid ECSA style.
- Our model and style specification emphasize quality constraints. We define most of the important quality attributes of ECSA. They can not only help analyzing and evaluating ECSA systems, but also guide designing ECSA architecture.

4. Conclusions and Future Work

SOA-centric enterprises are now extending the existing enterprise SOA architectures and investments into the Cloud to address new business pressure and opportunities. Cloud-centric enterprises are now adopting ESOA principles and governance for improving quality of cloud services and making cloud computing adopted by other enterprises. Therefore the research of how to make ESOA and CC work together with new SOA and CC hybrid architectural style is very important and drawing more attentions from both researchers and industry.

In this paper, we define a formula (2.2) of cloud service-oriented architectural style, which combines both ESOA style [27][29] and cloud computing style in a synergic way. The ECSA architectural style specifies the vocabulary of ECSA architectural elements; encapsulates important decisions about the ECSA architectural elements; and emphasizes important constraints on the elements and their relationships. Therefore, the research of ECSA can benefit both researchers and practitioners for better understanding ECSA architecture and making the right architectural decisions.

<u>Our major contributions are</u>

- Define and specify the ECSA by the formula (2.2). It gives an insight of design principles, structures and behaviors in both functional and non-functional of ECSA architectures.
- Propose a dynamic data center architecture.

The methodology and model proposed in the paper can be used for

- analyzing and evaluating broad-types of concrete enterprise architectures, and
- guiding the design of cloud-enable ESOA system or ESOA style cloud computing systems.

<u>Our future research includes</u>

- Application of cloud *service-oriented* formula (2.2),
- Tradeoff Analysis of ECSA architectural quality attributes,
- Classification of ECSA substyles to guide style selection,
- Architecture of automatic SLA and policy enforcement to ECSA system, and

- Formal methods of analyzing and verifying ECSA architectures.

References

[1] Amazon, EC2 SLA, http://aws.amazon.com/ec2-sla/

[2] Amazon, Auto Scaling and load balance, http://aws.amazon.com/autoscaling/

[3] Amazon.com, Instance Metadata, http://docs.amazonwebservices.com/AWSEC2/latest/DeveloperGuide/index.html?AESDG-chapter-instancedata.html

[4] Amazon, Amazon Web Services: Overview of Security Processes, 2009. http://awsmedia.s3.amazonaws.com/pdf/AWS_Security_Whitepaper.pdf

[5] T. Anstett, F. Leymann, R. Mietzner and S. Strauch, Towards BPEL in the Cloud: Exploiting Different Delivery Models for the Execution of Business Processes, IEEE Congress on Services, pp. 670-685, 2009

[6] J. Brodkin, Gartner: Seven cloud-computing security risks Data integrity, recovery, privacy and regulatory compliance are key issues to consider, Network World, 2009 at http://www.networkworld.com/news/2008/070208-cloud.html

[7] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg and I. Brandic, Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility, *Future Gener. Comput. Syst.*, Vol. 25, No. 6. (2009), pp. 599-616.

[8] D. Chappell and D. Berry, SOA – Ready for Primetime: The Next-Generation, Grid-Enabled Service-Oriented Architecture, SOA Magazine, Sept. 2007

[9] M. Creeger, Cloud Computing: An Overview, ACM Queue, 2009

[10] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pichin, S. Sivasubramanian and P. Vosshall, Dynamo: Amazon's Highly Available Key-value Store, SOSP'07, Oct. 14-17, USA, 2007

[11] J. Dong, R. Paul, and L.-J. Zhang, High Assurance Service-Oriented Architecture, IEEE Computer, Volume 41, Issue 8, Pages 22-23, August 2008.

[12] J. Dong, R. Paul, and L.-J. Zhang, High Assurance Services Computing, Springer, 2009.

[13] T. Dornemann, E. Juhnke and B. Freisleben, On-Demand Resource Provisioning for BPEL Workflow Using Amazon's Elastic Compute Cloud, The 9th IEEE/ACM International Symposium on Cloud Computing and the Grid, pp. 140-147, 2009.

[14] I. Foster, Y. Zhao, I. Raicu and S. Lu, Cloud Computing and Grid Computing 360-Degree Compared, *Grid Computing Environments Workshop, 2008.* pp. 1-10.

[15] J. Governor, D. Hinchcliffe and D. Nickull, Web 2.0 Architectures, Oreilly Media, May 2009

[16] Harvard Research Group, Cloud Computing – Introduction, http://www.hrgresearch.com/Cloud%20Computing.html

[17] HP, The Cloud and SOA, 2009, http://www.hp.com/hpinfo/analystrelations/wp_cloudcomputing_soa_capgemini_hp.pdf

[18] IBM, Seeding the Clouds: Key Infrastructure Elements for Cloud Computing, Feb. 2009. http://www-935.ibm.com/services/uk/cio/pdf/oiw03022usen.pdf

[19] G. Lakshman and P. Manish, How the Cloud Stretches the SOA Scope, p. 36-41, V. 21, Architecture Journal, 2009

[20] Loraine Lawson (blog), Identifying the Synergy Between SOA and the Cloud, *Mar 19, 2009,* http://www.itbusinessedge.com/cm/blogs/lawson/identifying-the-synergy-between-soa-and-the-cloud/?cs=31219

[21] D. S. Linthicum, Cloud Computing and SOA Convergence in Your Enterprise, Addison-Wesley, Sept. 2009.

[22] Microsoft, Window Azure Platform, 2009, http://www.microsoft.com/azure

[23] Oracle, Architectural Strategies for Cloud Computing, 2009,http://www.oracle.com/technology/architect/entarch/pdf/architectural_strategies_for_cloud_computing.pdf

[24] L. O'Brien, L. Bass and P. Merson, "Quality Attributes and Service-Oriented Architectures", Technical Note, CMU/SEI-2005-TN-014, 2005.

[25] G. Reese, Cloud Application Architectures, O'Reilly Media, Inc., 2009

[26] SUN, Introduction to Cloud Computing Architecture, 2009, http://www.sun.com/featured-articles/CloudComputing.pdf

[27] L. Tang, J. Dong and T. Peng, A Generic Model of Enterprise Service-Oriented Architecture, 4th IEEE International Symposium on Service-Oriented System Engineering (SOSE), pp 1-7, December 2008.

[28] L. Tang, Y. Zhao and J. Dong, Specifying Enterprise Web-Oriented Architecture, in High Assurance Services Computing, Springer, pages 241-260, 2009

[29] L. Tang, J. Dong, T. Peng and W. T. Tsai, Modeling Enterprise Service-Oriented Architectural Styles, Journal of Service Oriented Computing and Applications (SOCA). Springer, 2010.

[30] W. T. Tsai, Service-Oriented System Engineering: A New Paradigm, IEEE International Workshop on Service-Oriented System Engineering (SOSE), 2005, pp. 3-6

[31] L.-J. Zhang and Q. Zhou, CCOA: Cloud Computing Open Architecture, IEEE International Conference on Web Services, pp. 607-616, 2009.

[32] L.-J. Zhang, J. Zhang and H. Cai, Services Computing, Springer, Oct. 2007

[33] L.-J. Zhang, H. Li and H. Lam, Towards a Business Process Grid for Utility Computing, IT Professional, V. 6, N. 5, P. 63-64, 2004.

[34] Y. Zhao, J. Dong, and T. Peng, Ontology Classification for Semantic Web Based Software Engineering, IEEE Transactions on Services Computing (TSC). vol. 2, no. 4, pp. 303-317, Oct.-Dec. 2009.

[35] Jinesh Varia, Cloud Architecture, 2008, http://developer.amazonwebservices.com/connect/entry.jspa?externalID=1632

[36] Werner Vogels, Seamlessly Extending the Data Center - Introducing Amazon Virtual Private Cloud, Blogs, http://www.allthingsdistributed.com/2009/08/amazon_virtual_private_cloud.html

[37] Wikipedia, Google App Engine, http://en.wikipedia.org/wiki/Google_App_Engine