On the Communication Variability Analysis of the NeCTAR Research Cloud System

Zheng Li, *Member, IEEE*, Rajiv Ranjan, *Senior Member, IEEE*, Liam O'Brien, *Member, IEEE*,
He Zhang, *Member, IEEE*, Muhammad Ali Babar, Albert Y. Zomaya, *Fellow, IEEE*, and
Lizhe Wang, *Senior Member, IEEE*

***Abstract*—A national research Cloud is being created as part of the National eResearch Collaboration Tools and Resources (NeCTAR) project, to supply a cost-effective computational infrastructure to Australian scholars. Since it is hard to comprehensively understand the available Cloud resources through the open indicators, we decided to empirically investigate the infrastructural performance of the NeCTAR Research Cloud. In particular, our current evaluation work shows that the communication data throughput vary significantly within and between the five existing Cloud nodes. Although operated as one Cloud system, the NeCTAR Research Cloud might have employed heterogeneous network equipment and/or technologies. As the NeCTAR project is still ongoing, the one-off evaluation results could be quickly out of date, and the evaluation implementations should be frequently repeated or replicated for monitoring or consuming the Cloud services. Thus, unlike the existing evaluation studies that mainly focused on the evaluation results or tools, we emphasize the complete procedure and backend logic of evaluation implementations. Practically, as demonstrated in this paper, the procedure and logic can be recorded into Cloud Evaluation Experiment Methodology-based evaluation templates to facilitate repeatable and comparable Cloud services evaluation.**

***Index Terms*—Cloud computing, cloud services evaluation, communication variability, evaluation template, National eResearch Collaboration Tools and Resources (NeCTAR) research cloud.**

TABLE I
NECTAR RESEARCH CLOUD INSTANCES AVAILABLE BY SIZE

VM Type	CPU	Memory	Storage (secondary disk)
Small	1 core	4GB	30GB
Medium	2 cores	8GB	60GB
Large	4 cores	16GB	120GB
XLarge	8 cores	32GB	240GB
XXLarge	16 cores	64GB	480GB

I. INTRODUCTION

CLOUD computing has been widely accepted as a popular computing paradigm [1]. Although Cloud computing emerged as a business model, the researchers have increasingly regarded Cloud resources as a flexible and encouraging alternative to satisfy academic requirements [2]. In addition to exploiting the public Cloud, academia is also developing dedicated Clouds to facilitate research activities. For example, the National eResearch Collaboration Tools and Resources (NeCTAR) project is creating a federated research Cloud to cost-effectively supply computational infrastructures to the Australian research community [3]. This research Cloud will consist of up to eight nodes distributed around Australia. When employing service instances, the Cloud users are supposed to specify a suitable zone to get the best network responsiveness.

To satisfy potentially diverse application requirements, Cloud providers usually offer different types of Cloud services. In essence, the same Cloud infrastructure would have been virtualized as various resource types. For example, Amazon provides several options for its storage service, such as S3, EBS, and the local disk on EC2 [4]; while the NeCTAR Research Cloud varies the available resources into five instance sizes [3], as listed in Table I. As such, Cloud usage requires a deep understanding of how different candidate services may (or may not) match particular demands. However, consumers in general have little knowledge and control over the precise nature of Cloud services even in a "locked down" environment [5], and the given indicators often do not reflect comprehensive information about the overall performance of a service regarding specific tasks [6]. When it comes to the NeCTAR Research Cloud, in particular, besides the indicators of computation, memory, and storage indicators, NeCTAR does not even provide any communication indicator of its instances (cf. Table I). As such, to find the best network-responsive zone, evaluation would play a prerequisite role in employing the NeCTAR Cloud services.

Manuscript received April 26, 2015; revised August 13, 2015; accepted September 17, 2015. NICTA was supported by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program. NICTA was also supported by the Australian Capital Territory, the New South Wales, Queensland and Victorian Governments, the Australian National University, the University of New South Wales, the University of Melbourne, the University of Queensland, the University of Sydney, Griffith University, Queensland University of Technology, Monash University, and other university partners.

Z. Li is with the Department of Electrical and Information Technology, Lund University, 223 63 Lund, Sweden.

R. Ranjan is with Newcastle University, Tyne NE1 7RU, U.K., and Commonwealth Scientific and Industrial Research Organisation (CSIRO), Clayton, Vic. 3168, Australia.

L. O'Brien is with the ICT Innovation and Services, Geoscience Australia, Canberra, ACT 2609, Australia.

H. Zhang is with the State Key Laboratory of Novel Software Technology, Software Institute, Nanjing University, Nanjing 210093, China.

M. Ali Babar is with the School of Computer Science, University of Adelaide, Adelaide, SA 5005, Australia.

A. Y. Zomaya is with the School of Information Technologies, The University of Sydney, Sydney, NSW 2006, Australia.

L. Wang is with the School of Computer, China University of Geosciences, Wuhan 430074, China (e-mail: Lizhe.Wang@computer.org).

Digital Object Identifier 10.1109/JSYST.2015.2483759

1) **Requirement Recognition:** Recognize the problem and state the purpose of a proposed evaluation.
2) **Service Feature Identification:** Identify Cloud services and the features to be evaluated.
3) **Metrics and Benchmarks Listing:** List all the metrics and benchmarks that may be used for the proposed evaluation.
4) **Metrics and Benchmarks Selection:** Select suitable metrics and benchmarks for the proposed evaluation.
5) **Experimental Factors Listing:** List all the factors that may be involved in the evaluation experiments.
6) **Experimental Factors Selection:** Select limited factors to study, and also choose levels/ranges of these factors.
7) **Experimental Design:** Design experiments based on the above work. Pilot experiments may also be done in advance to facilitate the experimental design.
8) **Experimental Implementation:** Prepare experimental environment and perform the designed experiments.
9) **Experimental Analysis:** Statistically analyze and interpret the experimental results.
10) **Conclusions and Documentation:** Draw conclusions and report the overall evaluation procedure and results.

Fig. 1. Cloud Evaluation Experiment Methodology (CEEM) for Cloud services evaluation (cf. [10]).

In general, Cloud services evaluation is challenging, and the existing evaluation implementations are relatively incomplete and not well-organized. We have identified two main issues from the literature. First, many studies emphasized automated tools and benchmarks when evaluating Cloud services, which however do not necessarily guarantee rational and comparable evaluation results. On one hand, an ideal Cloud benchmark is still far from existence [7], and thus various traditional benchmarks have been largely employed in Cloud services evaluation. On the other hand, the same tool/benchmark can be adopted to evaluate different features of a Cloud service in different circumstances. For example, the benchmark Bonnie++ has been used for investigating both the memory and the storage performance of Amazon EC2 [8].

Second, some practitioners were concerned about evaluation results only, without caring about the corresponding evaluation procedures. In the domain of Cloud computing, however, previous evaluation results might become quickly out of date. Cloud providers may continually upgrade their hardware and software infrastructures, and new technologies may even be gradually employed. Moreover, the reported evaluation results could have been further flawed due to improper evaluation activities. For example, irrational and meaningless benchmarking results (e.g., higher-rank service performs worse) can be frequently seen in the CloudHarmony depository [9], while there are few clues for us to trace or replicate the original experiments.

Therefore, we decided to follow the Cloud Evaluation Experiment Methodology (CEEM, cf. Fig. 1) and concentrate on the evaluation workflow to perform systematic investigations into the performance of NeCTAR Research Cloud. In particular, we developed CEEM-based evaluation templates for traceable and reproducible evaluation implementations. In this paper, considering the aforementioned lack of communication indicator of the NeCTAR Research Cloud, we mainly focused on its communication performance by examining the inter-node and intra-node data throughput. The contributions of this work are threefold.

1) Our evaluation results can help developers of the NeCTAR Research Cloud to better understand the communication variability in/between the Cloud nodes, and accordingly improve the Cloud's infrastructural performance.
2) Our evaluation results can help consumers of the NeCTAR Research Cloud (i.e., the Australian research community, especially Canberra-based users in this case) to select communication-efficient Cloud resources for their network-intensive applications.
3) The whole evaluation logic and details reported in this paper can be viewed as a reusable instructions template of evaluating the NeCTAR Cloud communication. As such, future evaluations can be conveniently repeated or replicated based on this template, even by different evaluators at different times and locations. More importantly, given the regulated backend logic and the specific process, all the template-driven evaluation implementations would be traceable and comparable.

The remainder of this paper is organized as follows. Section II summarizes the existing work related to the communication evaluation of Cloud services. Section III briefly introduces the evaluation methodology CEEM employed in our study. Section IV specifies the logic and detailed activities driven by CEEM for evaluating the communication performance within and between the NeCTAR Research Cloud nodes. Conclusion and some future work are discussed in Section V.

II. Related Work

Unlike in-house computing resources, Cloud services normally have to be consumed through the Internet/Ethernet. Consequently, communication can be treated as an instinctive property of all types of Cloud services. Unfortunately, to the best of our knowledge, not many studies have dedicatedly investigated the communication performance of Cloud services. Wang and Ng [11] particularly focused on the impact of virtualization on the communication performance in a Cloud data center. Since virtualization techniques use a special privileged virtual machine (VM) (i.e., a driver domain) to control and allow the other VMs to access the physical network devices, they found that virtualization could bring about very unstable Communication Data Throughput, and different VMs might also experience communication variation due to their processor sharing. Considering this study was conducted in a single data center without measuring network performance between different data centers, Cerviño *et al.* [12] extended the original evaluation experiments and tested nine communication paths within and between three different geographical locations of Amazon EC2. However, the inconsistent evaluation activities (e.g., service feature identification and benchmark selection) made these two studies hardly comparable in terms of both evaluation implementations and evaluation results.

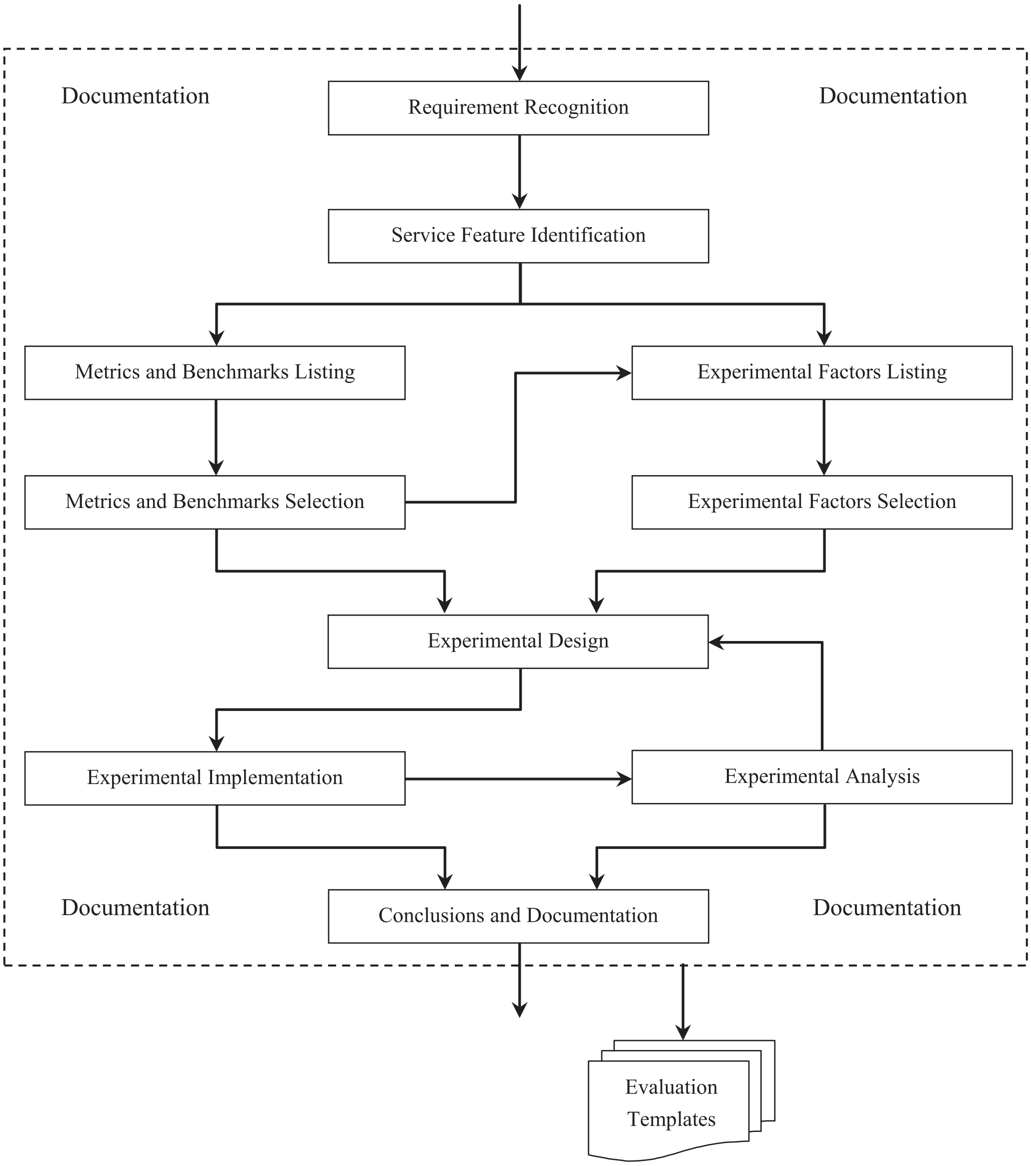

Documentation
Requirement Recognition
Documentation
Service Feature Identification
Metrics and Benchmarks Listing
Experimental Factors Listing
Metrics and Benchmarks Selection
Experimental Factors Selection
Experimental Design
Experimental Implementation
Experimental Analysis
Documentation
Conclusions and Documentation
Documentation
Evaluation Templates

Fig. 2. Overall logic with input-output data flow in CEEM.

In contrast, most relevant studies involved communication evaluation when investigating various Cloud service properties. For example, Li *et al.* were concerned about communication performance of Cloud services in their Cloud provider comparator CloudCmp [13]; Iosup *et al.* employed a wide variety of benchmarks to estimate the overall performance of public Cloud service resources, including communication [14]; and the network bandwidth was emphasized as one of six performance aspects when measuring the runtime of Cloud services [15]. However, these studies tended to report one-off evaluation results, while our work aims at repeatable and comparable evaluation implementations.

Furthermore, considering Message Passing Interface (MPI) programs are typical and common scientific applications, in addition to the IP-level communication, evaluators also considered the MPI-message-level networking of Cloud services. For instance, in the context of high performance computing, the Cloud has been frequently compared with an in-house cluster with regard to the MPI communication [16], [17]. Apparently, the MPI communication performance of the Cloud also needs to be reflected by using a cluster of VMs. In this paper, to simplify the demonstration, we only monitor the NeCTAR Research Cloud from the IP-level perspective.

III. Cloud Evaluation Experiment Methodology (CEEM)

To emphasize rigorous evaluation logic and follow a systematic evaluation procedure, we employ the aforementioned methodology CEEM in this study. Recall that a methodology refers to "an organized set of principles which guide action in trying to 'manage' (in the broad sense) real-world problem situations" [18]. CEEM organizes ten generic evaluation steps, as listed in Fig. 1. Each evaluation step has its own input, activities, strategies, and output.

By connecting the input and output between consecutive steps, the overall evaluation logic embedded in CEEM can be illustrated as outlined in Fig. 2. In detail, the procedure of Cloud services evaluation driven by CEEM can be roughly divided into two parts: the linear-process part related to pre-experimental activities and the spiral-process part related to experimental activities. In particular, the experimental activities would follow a spiral process, because an evaluation implementation could be composed of a set of experiments, while the experimental design in a later iteration could have to be determined by the experimental results and analyses from a prior iteration.

Furthermore, CEEM integrates a set of knowledge artefacts to facilitate conducting relevant evaluation activities. Benefiting from the specific logic and instructions, CEEM is supposed to help practitioners perform more traceable and reproducible evaluations, obtain more objective experimental results, and draw more convincing conclusions.

IV. Communication Evaluation

The NeCTAR Cloud system has been recognized to be a significant computational resource which complements existing and new supercomputing facilities in Australia. As one of the largest Openstack based Clouds in the world, this Research Cloud will consist of 30000+ cores of computing power distributed within up to eight nodes and run by bodies selected by NeCTAR. Ideally, end users would have a seamless experience regardless of which node within the federated Cloud they access. To get the best network responsiveness, however, the end users still have to identify the most suitable resource zone for their research projects. In other words, communication evaluation would be crucial for employing the NeCTAR Cloud service.

As mentioned previously, we followed CEEM (cf. Fig. 1) to systematically investigate the communication performance of the NeCTAR Research Cloud and also to highlight the whole procedure of evaluation implementation. As a use case of CEEM, in the following subsections, we mainly describe the CEEM-driven evaluation activities, while not elaborating the justification for those ten evaluation steps.

A. Requirement Recognition

Driven by CEEM, the first step is to discuss with the recipients of future evaluation results and achieve a clear statement of the purpose of Cloud services evaluation. Technically, evaluators create mapping between the elements in the performance evaluation *Taxonomy* [19] and the natural-language descriptions about the evaluation objectives, and then deliver a set of specific requirement questions to be addressed by potential evaluation experiments.

Since the NeCTAR Research Cloud employs a set of nodes to provide cost-effective computational infrastructures to the Australian academic community [3], a Canberra-based Cloud user may ask, “I would like to know which NeCTAR Cloud node is more suitable for my network-intensive application?” In this case, the Canberra-based Cloud user is the recipient of evaluation results. Given the keyword “network-intensive,” we used the taxonomy to define four specific requirement questions for this NeCTAR Cloud evaluation, such as:

1) How fast does the NeCTAR research Cloud transfer data internally?
2) How fast can data be transferred between the NeCTAR research Cloud and the Canberra-based user?
3) How variable does the NeCTAR research Cloud transfer data during a particular period of time?
4) How variable does the NeCTAR research Cloud transfer data via different communication paths?

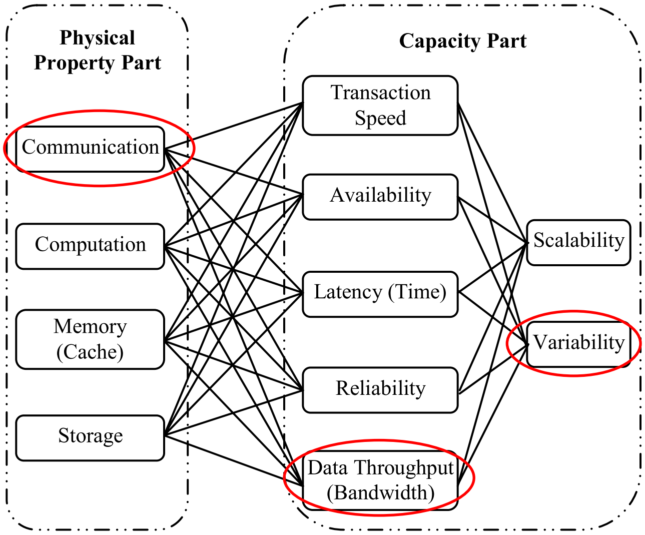


Fig. 3. Potential performance features for Cloud services evaluation (cf. [19]).

B. Service Feature Identification

Given a particular evaluation requirement, evaluators should identify the relevant Cloud services and their features to be evaluated. Since it is hard to outline the scope of Cloud Computing [20], and various Cloud services are increasingly available in the market [13], it could be difficult to directly locate proper features of a particular Cloud service. To facilitate service feature identification, CEEM provides a list of general Cloud service features through the aforementioned performance evaluation taxonomy. As such, evaluators can extract the Cloud resource- and capacity-related terms from each of the requirement questions, and then further refer to the candidate service features and select the suitable ones.

Here, we have identified two communication-related features of the NeCTAR Cloud service, as illustrated in Fig. 3. In particular, Variability of Communication Data Throughput can be viewed as a secondary service feature over the primary one, i.e., Communication Data Throughput. Note that different requirement questions may correspond to one service feature, with different experimental setup scenes.

1) Communication Data Throughput.
2) Variability of Communication Data Throughput.

TABLE II
CANDIDATE METRICS AND BENCHMARKS FOR EVALUATING COMMUNICATION DATA THROUGHPUT

Capacity	**Metrics**	**Benchmark**
Data Throughput	TCP/UDP/IP Transfer bit/Byte Speed (bps, Mbps, MB/s, GB/s)	iPerf [13]
		Private tools TCPTest/UDPTest [11]
		SPECweb 2005 [22]
		Upload/Download/Send large size data [23]
	MPI Transfer bit/Byte Speed (bps, MB/s, GB/s)	HPCC: b_eff [8]
		Intel MPI Bench [16]
		mpptest [24]
		OMB-3.1 with MPI [25]

TABLE III
CANDIDATE METRICS FOR EVALUATING VARIABILITY OF COMMUNICATION DATA THROUGHPUT

Sample	**Metrics**
[26]	Average, Minimum, and Maximum Value
[27]	Coefficient of Variation (COV) (ratio)
[23]	Difference between Min & Max (%)
[28]	Standard Deviation with Average Value
[29]	Cumulative Distribution Function Chart
[11]	Probability Density Function
[30]	Quartiles Chart with Median/Mean Value
[29]	Representation in Single Chart
[30]	Representation in Separate Charts
[29]	Representation in Table

C. Metrics and Benchmarks Listing

Before determining suitable metrics and benchmarks for evaluating particular Cloud service features, this step involves listing all the candidate options in advance. CEEM allows evaluators to use the identified Cloud service features as retrieval keys to quickly search metrics and benchmarks from our *Metrics Catalogue* [21]. Note that the *Metrics Catalogue* essentially provides a lookup capability for finding suitable metrics and benchmarks used in the existing experiences of evaluating Cloud services. Meanwhile, group meetings and expert opinions are not supposed to be completely replaced with the *Metrics Catalogue*. New metrics and benchmarks can still be supplemented by domain experts and other evaluators.

In this paper, we did not resort to expert judgment. By exploring the *Metrics Catalogue* for the service feature Communication Data Throughput, we listed the candidate metrics and benchmarks in Table II; as for the service feature Variability of Communication Data Throughput, the candidate metrics were listed in Table III. In fact, the Variability metrics are common for all the primary performance features including Communication Data Throughput.

D. Metrics and Benchmarks Selection

Given the previously listed candidates, the decision on metrics/benchmarks selection can be made by checking all the available resources, estimating the overhead of potential experiments, and judging the evaluators' capabilities of operating different benchmarks.

We chose Iperf as the benchmark because it has been identified to be able to deliver more precise results by consuming less system resources [15]. Moreover, Iperf can be run over cross-platform networks and generate standard performance measurements. Although the primary goal of Iperf is to help tune TCP connections over a particular network path, evaluators can use it to measure achievable bandwidth on IP networks within particular TCP window sizes.

Correspondingly, the selected metrics are:

1) Metric for the feature Communication Data Throughput.
   a) TCP Transfer bit Speed.

2) Metrics for the feature Variability of Communication Data Throughput.
   a) COV.
   b) Standard Deviation with Average Value.
   c) Representation in Table.
   d) Representation in Single Chart.

In particular, the two metrics *Standard Deviation with Average Value* and *Representation in Table* are used together to measure the time-related variability, the metric *COV* is to compare the time-related variability at different communication paths, while the metric *Representation in Single Chart* is used to visualize the location-related variability. Note that COV enables comparing the degree of variation between different data series, even if the data series have different means.

E. Experimental Factors Listing

Based on the identified Cloud service features and the selected metrics and benchmarks, the purpose of this step is to list all the candidate experimental factors that might affect the service features to be evaluated. In fact, it has been identified that knowing all factors (also called parameters or variables) that affect the service feature is a tedious but necessary task. Following CEEM, evaluators can screen and look up potential factors in our *Experimental Factor Framework* [31]. In particular, the identified service features are used to explore Cloud Resource factors, the selected benchmarks are used to search Workload factors, and the selected metrics are directly used as service Capacity factors.

For example, in this case, we listed the candidate resource factors related to NeCTAR Cloud communication as shown in Fig. 4, the workload-related factors are listed in Figs. 5 and 6 illustrate the factors representing the capacity of different service features. The detailed explanations of those factors are specified in [31].

F. Experimental Factor Selection

Given the list of candidate experimental factors, the evaluators can then select limited ones that are of most interest and also determine the values of the selected factors. Note that

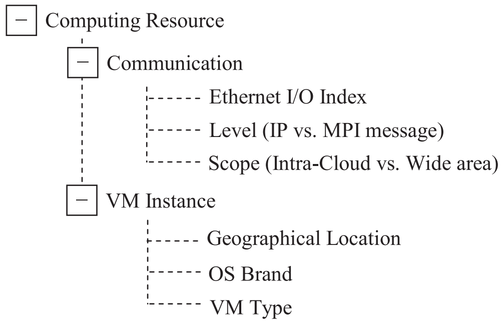


Fig. 4. Candidate factors related to the resource of NeCTAR Cloud communication.

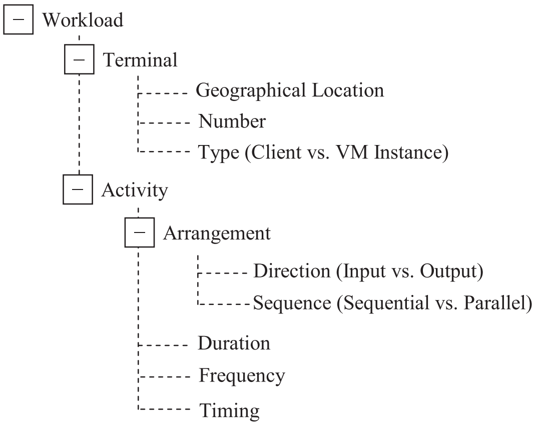

Workload
Terminal
Geographical Location
Number
Type (Client vs. VM Instance)
Activity
Arrangement
Direction (Input vs. Output)
Sequence (Sequential vs. Parallel)
Duration
Frequency
Timing

Fig. 5. Candidate factors related to the workload of using Iperf.

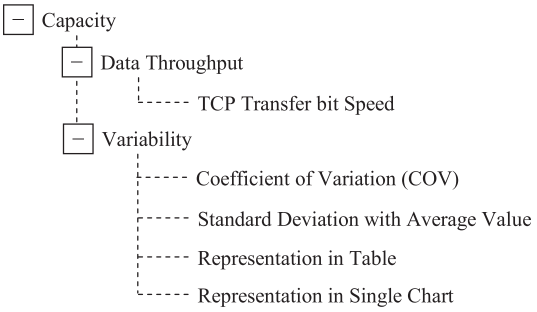

Capacity
Data Throughput
TCP Transfer bit Speed
Variability
Coefficient of Variation (COV)
Standard Deviation with Average Value
Representation in Table
Representation in Single Chart

Fig. 6. Candidate factors related to the capacity of two NeCTAR Cloud communication features.

there is no conflict between selecting limited factors in this step and keeping a comprehensive factor list in the previous step. On the one hand, an evaluation requirement usually comprises a set of experiments covering different factors. On the other hand, intentionally excluding unused factors does not mean that evaluators have not considered them.

The experimental factors can be selected by roughly considering pilot experiments. In detail, the evaluators can use the experimental setup scenes listed in our *Taxonomy* [19] to try different factor combinations for different experimental scenarios. Then, the suitable factors can be determined by selecting proper experimental scenarios. In fact, driven by separate requirement questions, CEEM naturally divides a whole evaluation into separate experimental scenarios, and each experimental scenario includes limited factors only.

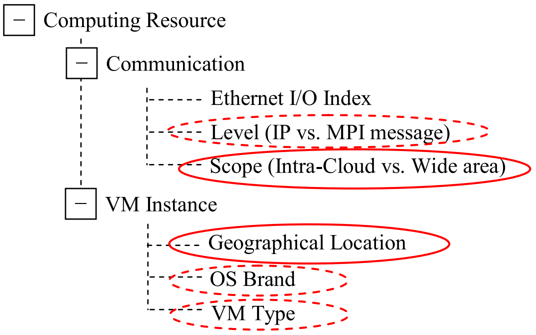


Fig. 7. Selected factors related to the resource of NeCTAR Cloud communication. (*The un-circled factors are intentionally excluded, and the dash-circled factors are considered for preparing an experimental environment instead of an experimental design.*)

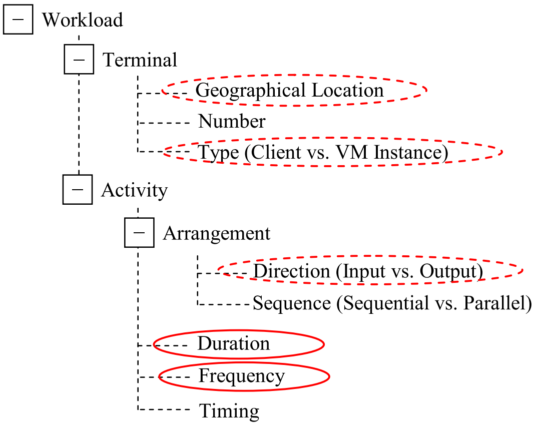

Workload
Terminal
Geographical Location
Number
Type (Client vs. VM Instance)
Activity
Arrangement
Direction (Input vs. Output)
Sequence (Sequential vs. Parallel)
Duration
Frequency
Timing

Fig. 8. Selected factors related to the workload of using Iperf. (*The un-circled factors are intentionally excluded, and the dash-circled factors are considered for preparing an experimental environment instead of an experimental design.*)

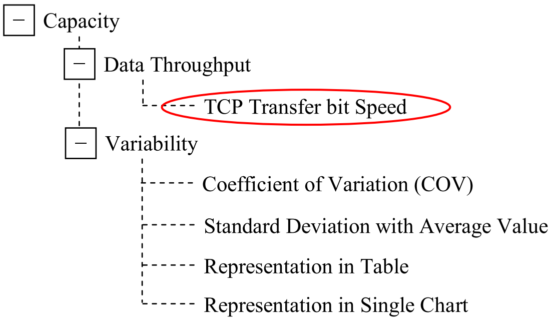

Capacity
Data Throughput
TCP Transfer bit Speed
Variability
Coefficient of Variation (COV)
Standard Deviation with Average Value
Representation in Table
Representation in Single Chart

Fig. 9. Selected factors related to the capacity of two NeCTAR Cloud communication features. (*The un-circled factors are intentionally excluded. Note that the selected capacity factor is essentially a metric that reflects output response to those input factors.*)

The Figs. 7–9 show the factors selected from the previous candidate list for the NeCTAR Cloud communication evaluation. In particular, Activity Duration and Frequency were factors for the time-related experiments, while Communication Scope and VM Geographical Location were factors for the location-related experiments, and we only considered TCP Transfer bit Speed as the response factor in all the experiments, because the variability metrics can be viewed as secondary calculations and visualizations based on TCP Transfer bit Speed.

The un-circled factors were intentionally excluded, and the dash-circled factors were considered for preparing an experimental environment instead of an experimental design. For example, when determining factor values, we chose m1.small as the cheapest VM type for this communication evaluation, and each VM installed the desktop system of Ubuntu 12.10 (Quantal) amd64, we employed a local machine in our NICTA Canberra Lab as the only client. The NeCTAR Cloud was always the server side when configuring the benchmark Iperf.

G. Experimental Design

Once the input-process variables (experimental factors) and output-process responses (metrics) have been selected, evaluation experiments need to be subsequently prepared and designed. Normally, a small scale of pilot experiments would benefit the relevant experimental design. For example, the trial runs of an evaluation experiment may help evaluators get familiar with the experimental environment, optimize the experimental sequence, and so on.

The designed experiments can be further characterized and recorded by building UML-style *Experimental Blueprints* [32]. For complex evaluation projects comprising collaboration between multiple evaluators, characterizing the designed experiments is particularly helpful to facilitate information exchanging and to avoid experimental duplications. Therefore, as a supplement to specific experimental instructions, the evaluators can use experimental blueprints to facilitate discussions among different people or to facilitate “apple-to-apple” comparisons between different evaluation experiments.

When designing experiments for the NeCTAR Cloud communication evaluation, we took into account time- and location-related factors separately. The time-related factors include *Activity Duration* (i.e., repeating experimental activity for a period of time) and *Activity Frequency* (i.e., repeating experimental activity for a number of times). In this case, we considered *Activity Duration*, and planned to keep Iperf running for 12 h. Since this design is a continuous observation, there is no need to further decide experimental sample sizes.

From the Cloud service’s perspective, the location-related factor indicates variable places where Cloud data centers are hosted. In fact, for reasons like disaster recovery, Cloud providers usually deploy many data centers in different geographical locations. Here, we were concerned with the five available NeCTAR Cloud nodes (namely MEB-np, MEB-qh2, Monash01, qld, and sa) at the time of writing. Thus, there are 15 $(= C(5,2) + C(5,1)$, inter-node plus intra-node) possible intra-Cloud communication paths, and five $(= C(5,1))$ possible wide-area communication paths between the NICTA client and the NeCTAR Cloud. The overall (output) location-related design is listed in Table IV, where each X indicates a communication path that needs to be tested.

In addition, we drew an experimental blueprint to characterize the designed experiments, as shown in Fig. 10. Note that the characterization is on an abstract level. The blueprint would still be the same even if the values of the relevant factors were changed, for example, running Iperf for a longer time or involving more Cloud nodes.

TABLE IV
LOCATION-RELATED EXPERIMENTAL DESIGN FOR THE NECTAR CLOUD COMMUNICATION EVALUATION

Path	MEB-np	MEB-qh2	Monash01	qld	sa
Local Machine	X	X	X	X	X
MEB-np	X	X	X	X	X
MEB-qh2		X	X	X	X
Monash01			X	X	X
qld				X	X
sa					X

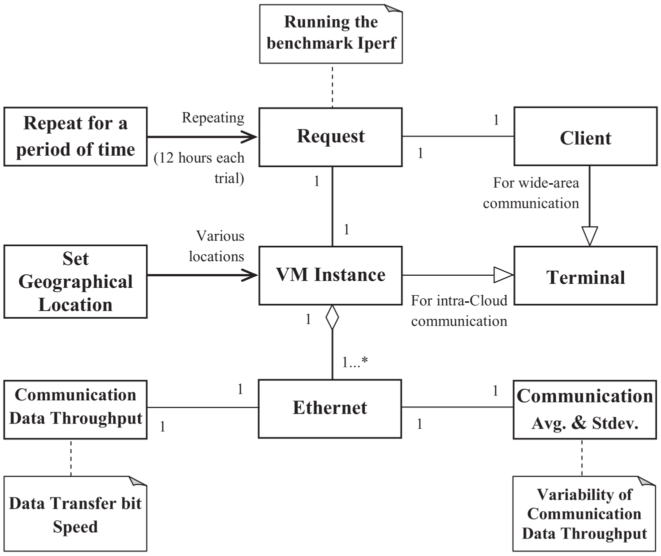

Running the benchmark Iperf
Repeat for a period of time
Repeating
(12 hours each trial)
Request
1
1
Client
1
For wide-area communication
1
Set Geographical Location
Various locations
VM Instance
1
Terminal
For intra-Cloud communication
1
1...*
Communication Data Throughput
1
1
Ethernet
1
1
Communication Avg. & Stdev.
Data Transfer bit Speed
Variability of Communication Data Throughput

Fig. 10. Experimental blueprint for the NeCTAR Cloud communication evaluation.

H. Experimental Implementation

CEEM requires experimental implementation to rigorously follow its corresponding design. To reach the rigorousness, evaluators can make the experimental actions as automated as possible to increase repeatability and reduce human mistakes. Note that CEEM regards pilot experimental runs as the evaluation activities in *Experimental Design* instead of in this step.

```
#!/bin/sh
for i in $ (seq 1 5000)
do
   echo “$i th running!/n”
   yourdate=′ date + %Y-%m-%d-%H:%M:%S′
   echo $yourdate ≫ iperf_result.txt
   iperf -c xxx.xxx.xxx.xxx -t 30 ≫
                       iperf_result.txt
   echo “/n sleeping.../n”
   sleep 30
done
echo “Done!”
```

TABLE V
EXPERIMENTAL RESULTS OF THE NECTAR CLOUD COMMUNICATION EVALUATION

Data Throughput (Mbits/sec) [Average (standard deviation)]	**MEB-np**	**MEB-qh2**	**Monash01**	**qld**	**sa**
Local Machine	20.2 (0.24)	19.9 (1.23)	19 (1.62)	14.8 (1.11)	11.6 (0.92)
MEB-np	5212.1 (458.3)	3799.5 (832.6)	2938.5 (302.2)	364.8 (41.97)	98.8 (9.61)
MEB-qh2		4496.8 (783.1)	1743.9 (517.4)	18.4 (9.52)	98.5 (7.05)
Monash01			2750.7 (138.8)	215.5 (95)	93.7 (6.41)
qld				2632.5 (160.4)	229.5 (58.2)
sa					5222.4 (417.8)

In fact, we have developed a suite of codes and scripts to automate preparing experimental environments and driving benchmarks for the NeCTAR Cloud communication evaluation. In this step, we further extended the codes and scripts for driving experimental implementation. For example, we developed the shell script for continuously running Iperf within the Linux environment, as shown above. Note that the “xxx.xxx.xxx.xxx” indicates a particular IP address of a Cloud instance on the server side.

After conducting all the experiments following the previous design, we obtained the experimental results and kept them in spreadsheets for future analysis.

I. *Experimental Analysis*

In this step, evaluators statistically analyze the experimental results where necessary and also visualize the analysis results if possible. Being compatible with the traditional evaluation lessons, CEEM also emphasizes that visualizing experimental results by using various graphical tools may significantly facilitate data analysis and interpretation. In fact, CEEM directly treats suitable charts as metrics for measuring Cloud services in our *Metrics Catalogue*.

By using a Variability metric to measure the experimental results, we initially answered the requirement questions, i.e., the average data throughput and the corresponding standard deviation via different communication paths, as listed in Table V.

Since the NeCTAR Cloud communication has different average data throughput via different paths over the same time, the standard deviations representing time-related variability in Table V are not directly comparable against each other. As mentioned previously, we further realized the comparison by normalizing these standard deviations (namely COV) of the Communication Data Throughput, as calculated by (1). We also visualized the comparison in Fig. 11.

$$\text{COV} = \frac{\text{Standard Deviation}}{\text{Average}}. \tag{1}$$

By visual inspection, it can be seen that the communication between MEB-qh2 and qld varies significantly over time, while the remote access to the MEB-np node is the most stable path for Canberra-based users.

As for the location-based variability, considering there are both inter-node and intra-node communications in the NeCTAR Cloud, we employed two schemes of normalization to make the visualized result more intuitive. In detail, for inter-node communications, the longer distance (cf. Fig. 12) indicates the lower data throughput between two data center nodes. The normalization of distance is defined by

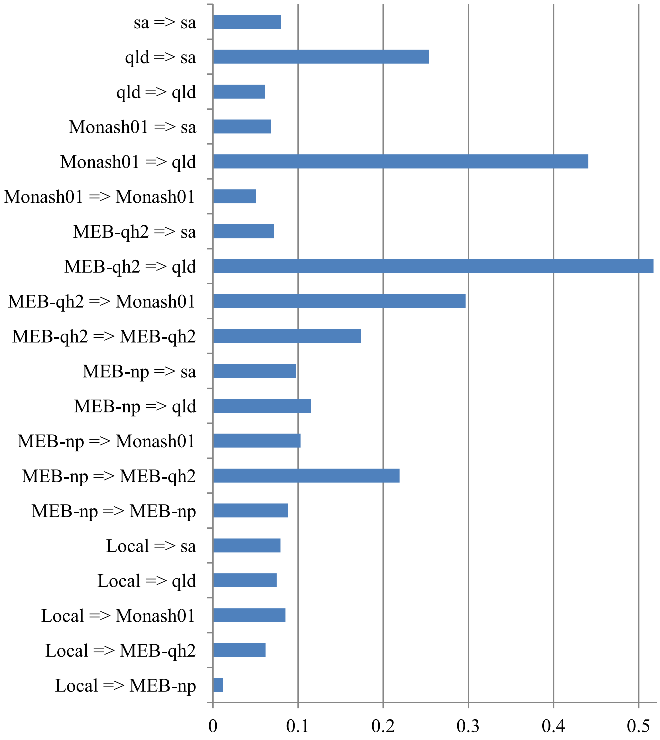


Fig. 11. Coefficent of Variation (COV) of data throughput at different communication paths over 12 h.

$$\text{Distance} = \frac{\text{MAX}(\text{Bandwidth}_{\text{inter}})}{\text{Bandwidth}_{\text{inter}}} \times \text{Ratio}_{\text{inter}}. \tag{2}$$

For intra-node communications, we used a larger circle (cf. Fig. 12) to indicate the higher data throughput within the corresponding data center node. The normalization of diameter is defined by

$$\text{Diameter} = \frac{\text{Bandwidth}_{\text{intra}}}{\text{MIN}(\text{Bandwidth}_{\text{intra}})} \times \text{Ratio}_{\text{intra}}. \tag{3}$$

Then, the average data throughput via different communication paths were visualized as shown in Fig. 12. The major findings through visual interpretation are: 1) From the perspective of our local machine, the sa node is the remotest Cloud node with the lowest data throughput. 2) There seem to be two

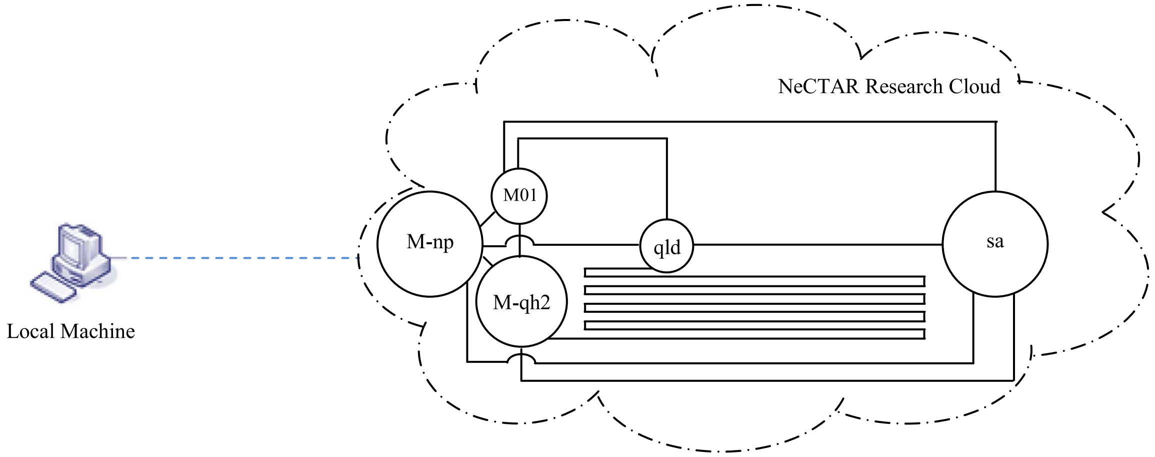


Fig. 12. Communication Data Throughput in the NeCTAR Research Cloud (M-np: the MEB-np node; M-qh2: the MEB-qh2 node; M01: the Monash01 node; qld: the qld node; sa: the sa node).

different patterns of the network infrastructure in the NeCTAR Research Cloud. The network infrastructure employed by the MEB-np, MEB-qh2, and sa nodes is nearly two times faster in terms of intra-node communication than the Monash01 and qld nodes. 3) It is clear that the communication between the MEB-qh2 and qld nodes is exceptionally slow compared to the other paths.

J. Conclusion and Documentation

CEEM uses a structured manner to draw evaluation conclusions and implement documentation. When drawing conclusions, evaluators can directly use tables and visualized representations of experimental (analysis) results to respond to the predefined requirement questions. Note that one chart or table might be used to answer different requirement questions, while the answer to one requirement question may comprise multiple charts and/or tables. Moreover, the answers to all the requirement questions can be further summarized into natural-language findings to better reflect the conclusions.

In addition to drawing conclusions, it is worth paying more attention to reporting the whole Cloud services evaluation implementations, because complete evaluation reports would be vital for other people to learn from or replicate/confirm previous evaluation practices. Following the steps of CEEM, evaluators can gradually document the evaluation details into a live maintained log. Such a live maintained log can then be used to generate structured evaluation reports and evaluation templates. In particular, the evaluation report mainly focuses on the whole logic of the evaluation procedure in natural language, while the evaluation templates mostly record the detailed environmental information, experimental instructions, and automated experimental actions to facilitate evaluation replication.

In this case, to begin with, we built links between suitable experimental (analysis) results and the corresponding requirement questions, as listed below.

1) How fast does the NeCTAR research Cloud transfer data internally?
   a) In the context of running the benchmark Iperf, the communication data throughput within the same Cloud node can be 2632.5 Mbits/s to 5222.4 Mbits/s, which is generally faster than that between different Cloud nodes, as listed in Table V.
2) How fast can data be transferred between the NeCTAR research Cloud and the Canberra-based user?
   a) In the context of running the benchmark Iperf, the communication data throughput between the Canberra-based user and three Melbourne-area Cloud nodes is around 20 Mbits/s, which is generally faster than that between the Canberra-based user and the other Cloud nodes, as listed in Table V.
3) How variable does the NeCTAR research Cloud transfer data during a particular period of time?
   a) Five communication paths (between different Cloud nodes) have relatively higher variation in the data throughput over time, while the others are relatively stable with COV less than 0.2, as shown in Table V and Fig. 11.
4) How variable does the NeCTAR research Cloud transfer data via different communication paths?
   a) As listed in Table V and illustrated in Fig. 12, the MEB-np, MEB-qh2, and sa nodes have nearly two times faster internal communication data throughput than the other two nodes; and the Melbourne-area Cloud nodes can transfer data between each other much faster compared to the other communication paths.

To summarize, we concluded: 1) For Canberra-based users, the Cloud resources located in Melbourne would be faster for remote access; 2) the MEB-np, MEB-qh2, and sa Cloud nodes could be more efficient for intra-node data replication and backup, while the three Melbourne nodes would be more friendly for inter-node operations; and 3) there could be something wrong with the network infrastructure between the MEB-qh2 (M-qh2) and qld nodes, and this line's communication performance should be significantly improved.

As for the documentation, what is outlined in this section can be viewed as a polished version of the live maintained structured log. In other words, we use this paper to demonstrate how we recorded CEEM-driven evaluation activities. In practice, most content of the natural-language descriptions in this log can be directly used to build an evaluation report for the Canberra-based users, while the specific evaluation instructions together with the developed codes/scripts can compose one or two CEEM-based evaluation templates to help monitor and track the communication performance of the NeCTAR Research Cloud, possibly by different people at different locations and times.

V. Conclusions and Future Work

Given the benefits such as on-demand capability and rapid elasticity of the emerging Cloud technology, both industry and academia are keen to take advantage of Cloud computing. A typical instance is the NeCTAR Research Cloud that is being created to supply cost-effective computing and computer power to researchers in Australia. Since the given indicators are far from enough to help comprehensively understand the available resources, we conducted empirical investigations into the infrastructural performance of the NeCTAR Research Cloud. In particular, we mainly focused on the Cloud's communication property. Our evaluation results show that, although operating as one Cloud system, the Communication Data Throughput varies significantly within and between the five existing Cloud nodes. The detailed experiments not only revealed communication-efficient Cloud resources from the consumers' perspective but also indicated directions for improving the NeCTAR Research Cloud from the developers' perspective. For example, in addition to the possible connection issue between the MEB-qh2 and qld nodes, we also identified that there could be heterogeneous network infrastructures employed in the NeCTAR Research Cloud.

More importantly, our evaluation methodology CEEM can be applied by Cloud providers on a day-to-day basis for improving their network provisioning. Driven by CEEM, evaluators can record their evaluation practices through developing and delivering CEEM-based evaluation templates. The CEEM-based evaluation templates would in turn facilitate repeatable and comparable monitoring of the network infrastructure even by different evaluators at different times and locations. For example, along with the development of the NeCTAR project, there is no doubt that it would be worth keeping an eye on the changes in its infrastructure. Considering one-off evaluation results cannot satisfy such a requirement in this case, we particularly emphasize the replicable and repeatable backend logic of evaluation implementations. As such, the NeCTAR engineers can conveniently reuse the same evaluation template to generate traceable and comparable evaluation results whenever necessary.

The limitation of this evaluation work is that we were only concerned with the traditional network architecture. Considering that Software-defined Network (SDN) has increasingly become a significant architecture for modern computing paradigms (e.g., to deal with today's big data challenges in massive parallel processing on thousands of servers), we will extend our focus to CEEM-driven evaluation of SDN Cloud infrastructures. Furthermore, our future work will be unfolded toward two directions. On the one hand, we will keep investigating the performance of various service features of the NeCTAR Research Cloud and continue developing corresponding evaluation templates. On the other hand, we will gradually make the evaluation templates public and use them to build a systematic standard for monitoring the Cloud infrastructures.

References

[1] 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, pp. 599–616, Jun. 2009.

[2] Z. Li, H. Zhang, L. O'Brien, R. Cai, and S. Flint, "On evaluating commercial cloud services: A systematic review," *J. Syst. Softw.*, vol. 86, no. 9, pp. 2371–2393, Sep. 2013.

[3] "Nectar research cloud," NeCTAR, Melbourne, Vic., Australia, 2014. [Online]. Available: https://www.nectar.org.au/research-cloud

[4] D. Chiu and G. Agrawal, "Evaluating caching and storage options on the amazon web services cloud," in *Proc. 11th IEEE/ACM Int. Conf. GRID*, Brussels, Belgium, Oct. 25–28, 2010, pp. 17–24.

[5] W. Sobel *et al.*, "Cloudstone: Multi-platform, multi-language benchmark and measurement tools for web 2.0," in *Proc. 1st Workshop CCA*, Chicago, IL, USA, Oct. 22–23, 2008, pp. 1–6.

[6] A. Lenk, M. Menzel, J. Lipsky, S. Tai, and P. Offermann, "What are you paying for?: Performance benchmarking for Infrastructure-as-a-Service," in *Proc. IEEE Int. Conf. CLOUD*, Washington, DC, USA, Jul. 4–9 2011, pp. 484–491.

[7] C. Binnig, D. Kossmann, T. Kraska, and S. Loesing, "How is the weather tomorrow?: Towards a benchmark for the Cloud," in *Proc. 2nd Int. Workshop DBTest*, New York, NY, USA, Jun. 29, 2009, pp. 1–6.

[8] S. Ostermann *et al.*, "A performance analysis of EC2 cloud computing services for scientific computing," in *Proc. 1st Int. Conf. Cloud-Comput.*, Munich, Germany, Oct. 19–21, 2009, pp. 115–131.

[9] "CloudHarmony: Transparency for the cloud," CloudHarmony, Laguna Beach, CA, USA, 2014. [Online]. Available: https://www.cloudharmony.com

[10] Z. Li, L. O'Brien, and H. Zhang, "CEEM: A practical methodology for cloud services evaluation," in *Proc. IEEE 9th World Congr. SERVICES*, Santa Clara, CA, USA, Jun. 28–Jul. 3, 2013, pp. 44–51.

[11] G. Wang and T. S. E. Ng, "The impact of virtualization on network performance of amazon EC2 data center," in *Proc. IEEE INFOCOM*, San Diego, CA, USA, Mar. 14–19, 2010, pp. 1–9.

[12] J. Cerviño, P. Rodríguez, I. Trajkovska, A. Mozo, and J. Salvachúa, "Testing a cloud provider network for hybrid P2P and cloud streaming architectures," in *Proc. IEEE Int. Conf. CLOUD*, Washington, DC, USA, Jul. 4–9, 2011, pp. 356–363.

[13] A. Li, X. Yang, S. Kandula, and M. Zhang, "CloudCmp: Comparing public cloud providers," in *Proc. 10th Annu. IMC*, Melbourne, Vic., Australia, Nov. 1–3, 2010, pp. 1–14.

[14] A. Iosup *et al.*, "Performance analysis of cloud computing services for many-tasks scientific computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 6, pp. 931–945, Jun. 2011.

[15] J. Schad, J. Dittrich, and J.-A. Quiané-Ruiz, "Runtime measurements in the cloud: Observing, analyzing, and reducing variance," *Proc. Very Large Data Base Endowment*, vol. 3, no. 1/2, pp. 460–471, Sep. 2010.

[16] Z. Hill and M. Humphrey, "A quantitative analysis of high performance computing with Amazon's EC2 infrastructure: The death of the local cluster?" in *Proc. 10th IEEE/ACM Int. Conf. GRID*, Banff, AB, Canada, Oct. 12–16, 2009, pp. 26–33.

[17] Y. Zhai, M. Liu, J. Zhai, X. Ma, and W. Chen, "Cloud versus in-house cluster: Evaluating amazon cluster compute instances for running MPI applications," in *Proc. Int. Conf. High Perform. Comput. Netw. Storage Anal. SC*, Seattle, WA, USA, Nov. 12–18, 2011, pp. 1–10.

[18] P. Checkland and J. Scholes, *Soft Systems Methodology in Action*. New York, NY, USA: Wiley, Sep. 1999.

[19] Z. Li, L. O'Brien, R. Cai, and H. Zhang, "Towards a taxonomy of performance evaluation of commercial cloud services," in *Proc. IEEE 5th Int. Conf. CLOUD*, Honolulu, HI, USA, Jun. 2012, pp. 344–351.

[20] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: State-of-the-art and research challenges," *J. Internet Serv. Appl.*, vol. 1, no. 1, pp. 7–18, May 2010.

[21] Z. Li, L. O'Brien, H. Zhang, and R. Cai, "On a catalogue of metrics for evaluating commercial Cloud services," in *Proc. 13th ACM/IEEE Int. Conf. GRID*, Beijing, China, Sep. 20–23, 2012, pp. 164–173.

[22] H. Liu and S. Wee, "Web server farm in the Cloud: Performance evaluation and dynamic architecture," in *Proc. 1st Int. Conf. CloudCom*, Beijing, China, Dec. 1–4, 2009, pp. 369–380.

[23] S. Hazelhurst, "Scientific computing using virtual high-performance computing: A case study using the amazon elastic computing cloud," in *Proc. Annu. Res. Conf. SAICSIT IT Res. Devel. Countries, Riding Wave Technol.*, Wilderness, South Africa Oct. 6–8, 2008, pp. 94–103.

[24] Q. He, S. Zhou, B. Kobler, D. Duffy, and T. Mcglynn, "Case study for running HPC applications in public Clouds," in *Proc. 1st Workshop ScienceCloud Conjunction 19th ACM Int. Symp. HPDC*, New York, NY, USA, Jun. 21, 2010, pp. 395–401.

[25] C. Evangelinos and C. N. Hill, "Cloud computing for parallel scientific HPC applications: Feasibility of running coupled atmosphere-ocean climate models on Amazon's EC2," in *Proc. IEEE 1st Workshop CCA*, Chicago, IL, USA, Oct. 22–23 2008, pp. 1–6.

[26] C. Baun and M. Kunze, "Performance measurement of a private cloud in the openCirrus testbed," in *Proc. 4th Workshop VHPC*, Delft, The Netherlands, Aug. 25, 2009, pp. 434–443.

[27] J. Dejun, G. Pierre, and C. H. Chi, "EC2 performance analysis for resource provisioning of service-oriented applications," in *Proc. 7th Int. Conf. ICSOC-ServiceWave*, Stockholm, Sweden, Nov. 23–27, 2009, pp. 197–207.

[28] Z. Hill, J. Li, M. Mao, A. Ruiz-Alvarez, and M. Humphrey, "Early observations on the performance of Windows Azure," in *Proc. 1st Workshop ScienceCloud Conjunction 19th ACM Int. Symp. HPDC*, Chicago, IL, USA, Jun. 21 2010, pp. 367–376.

[29] S. L. Garfinkel, "An Evaluation of Amazon's Grid Computing Services: EC2, S3, and SQS," Center Res. Comput. Soc., School Eng. Appl. Sci., Harv. Univ., Cambridge, MA, USA, Tech. Rep. TR-08-07, 2007.

[30] A. Iosup, N. Yigitbasi, and D. Epema, "On the performance variability of production Cloud services," in *Proc. 11th IEEE/ACM Int. Symp. CCGrid*, Newport Beach, CA, USA, May 23–26, 2011, pp. 104–113.

[31] Z. Li, L. O'Brien, H. Zhang, and R. Cai, "A factor framework for experimental design for performance evaluation of commercial cloud services," in *Proc. 4th IEEE Int. Conf. CloudCom*, Taipei, Taiwan, Dec. 3–6, 2012, pp. 169–176.

[32] Z. Li, L. O'Brien, H. Zhang, and R. Cai, "On the conceptualization of performance evaluation of IaaS services," *IEEE Trans. Serv. Comput.*, vol. 7, no. 4, pp. 628–641, Oct.–Dec. 2014.

Zheng Li (M'15) received the B.Eng. degree from Zhengzhou University, Zhengzhou, China; the M.Sc.Eng. degree from Beijing University of Chemical Technology, Beijing, China; the M.E. by Research degree from the University of New South Wales, Kensington, Australia; and the Ph.D. degree from the Australian National University, Acton, Australia.

Before moving abroad, he had around four-year industrial experience in China after finishing his Bachelor's and Master's degrees. During his Master's and Ph.D. studies in Australia, he was a Graduate Researcher with the Software Systems Research Group, National ICT Australia. He is currently a Postdoctoral Researcher with the Department of Electrical and Information Technology, Lund University, Lund, Sweden. His research interests include cloud computing, performance engineering, empirical software engineering, software cost/effort estimation, and web service composition.

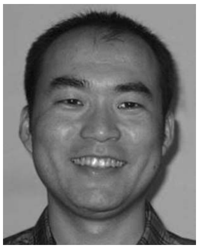
Rajiv Ranjan (SM'15) received the Ph.D. degree in computer science and software engineering from The University of Melbourne, Parkville, Australia, in 2009.

Previously, he was a Julius Fellow (2013–2015), a Senior Research Scientist (equivalent to Senior Lecturer in the Australian/U.K. University Grading System), and a Project Leader with the Digital Productivity and Services Flagship, Commonwealth Scientific and Industrial Research Organization. He is currently a Reader (Associate Professor) in computing science with Newcastle University, Newcastle upon Tyne, U.K., where he is working on projects related to emerging areas in parallel and distributed systems (i.e., cloud computing, the Internet of Things, and big data). He has published more than 140 peer-reviewed scientific papers, and his papers have received more than 4700 Google Scholar citations.

Liam O'Brien (M'14) received the Ph.D. degree from the University of Limerick, Limerick, Ireland, in 1996.

He has more than 25 years of experience in research and development in software engineering. Previously, he was with Lero, Limerick, and with Carnegie Mellon Software Engineering Institute, Pittsburgh, PA, USA. He was previously the Chief Software Architect with the Commonwealth Scientific and Industrial Research Organization and a Principal Researcher with the National ICT Australia's e-Government Initiative. He is currently an Enterprise Solutions Architect with Geoscience Australia, Canberra, Australia. He has published more than 60 peer-reviewed research papers in international journals, conferences, and workshops. His main areas of research include enterprise, systems and software architecture, service-oriented architecture, cloud computing, service science, software reuse, and software modernization.

Dr. O'Brien is a member of the Australian Computer Society and the Service Science Society of Australia.



He Zhang (M'12) Dr. O'Brien is a member of the Australian Computer Society and the Service Science Society of Australia.

He joined academia after seven years in industry, developing software systems in the areas of aerospace and complex data management. He is currently a Professor of software engineering with the Software Institute, Nanjing University, Nanjing, China. He has published more than 100 peer-reviewed research papers in international journals, conferences, and workshops. He undertakes research in software engineering, particularly software process (modeling, simulation, analytics, and improvement), software quality, empirical and evidence-based software engineering, service-oriented computing, software engineering research methodologies, etc.

Apart from being on the program committees and/or organizing committees of several international conferences such as the International Symposium on Empirical Software Engineering and Measurement, the International Conference on Software and System Process (ICSSP), the International Conference on Evaluation and Assessment in Software Engineering (EASE), and the International Conference of Product Focused Software Development and Process Improvement (PROFES), Dr. Zhang is also chairing the program committees of ICSSP 2013/2014, PROFES 2010, EASE 2015, and a number of workshops.

Muhammad Ali Babar was a Reader in software engineering with Lancaster University, Lancaster, U.K., and he was a Researcher and Project Leader in different research centers in Ireland and Australia. He is currently a Professor of software engineering (Chair) with the School of Computer Science, The University of Adelaide, Adelaide, Australia. He also holds an Associate Professorship at the IT University of Copenhagen, Copenhagen, Denmark. He has authored or coauthored more than 140 peer-reviewed research papers in journals, conferences, and workshops. He has coedited a book titled *Software Architecture Knowledge Management: Theory and Practice*.

Apart from being on the program committees of several international conferences such as the IEEE/IFIP Conference on Software Architecture and European Conference on Software Architecture, the International Symposium on Empirical Software Engineering and Measurement, the International Software Product Line Conference, the International Conference on Global Software Engineering, and the International Conference on Software and System Process for several years, Prof. Ali Babar was the founding General Chair of the Nordic-Baltic Symposium on Cloud Computing and Internet Technologies (NordiCloud) 2012. He has been a Guest Editor of several special issues/sections of IEEE Software, the *Journal of Systems and Software, Empirical Software Engineering Journal, Software and Systems Modeling, Information and Software Technology,* and *Requirements Engineering Journal*.

Albert Y. Zomaya (F'04) is currently the Chair Professor of High Performance Computing and Networking with the School of Information Technologies, The University of Sydney, Sydney, Australia. He is also the Director of the Centre for Distributed and High Performance Computing, which was established in late 2009. He published more than 500 scientific papers and articles and is the author, coauthor, or editor of more than 20 books. His research interests are in the areas of parallel and distributed computing and complex systems.

Prof. Zomaya is a Chartered Engineer and a Fellow of the American Association for the Advancement of Science and Institution of Engineering and Technology. He served as the Editor-in-Chief of the IEEE TRANSACTIONS ON COMPUTERS (2011–2014), and he also serves as an Associate Editor for 22 leading journals. He was the recipient of the IEEE Technical Committee on Parallel Processing Outstanding Service Award (2011), the IEEE Technical Committee on Scalable Computing Medal for Excellence in Scalable Computing (2011), and the IEEE Computer Society Technical Achievement Award (2014).

Lizhe Wang (SM'12) received the B.E. and M.E. degrees from Tsinghua University, Beijing, China, and the Doctor of Engineering degree (*magna cum laude*) from Karlsruhe Institute of Technology, Karlsruhe, Germany.

He is currently a Professor with the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing, and a Chutian Chair Professor with the School of Computer Science, China University of Geosciences, Beijing. His main research interests include high-performance computing, e-Science, and spatial data processing.

Prof. Wang is a Fellow of the Institution of Engineering and Technology and the British Computer Society. He serves as an Associate Editor of the IEEE TRANSACTIONS ON COMPUTERS and the IEEE TRANSACTIONS ON CLOUD COMPUTING.