The cloud computing paradigm has grown in popularity in industry and academia in recent years. Cloud computing aggregates several related technologies, such as virtualization, dynamic application scaling, datacenter management, and resource provisioning to provide an on-demand and elastic hosting environment for its users. While data management and storage are challenges, resource management is another significant challenge in the cloud environment, particularly when coupled with information management. The cloud environment has significant user constraints, such as multitenant, highly virtualized, and dynamic scaling. Effective resource management not only satisfies the resource requirements of user applications, but also optimizes system utilization so that more applications can be supported by the cloud at a satisfactory level.

As more users switch from the traditional, on-premises IT infrastructure to cloud-oriented platforms, cloud providers must address even more challenging and nuanced business requirements. Not only will cloud environments support users that lease virtual machines (VMs) to host straightforward legacy programs — they also will need to host complicated multistep applications, such as service workflows (that is, business or scientific processes where different services coordinate together to accomplish a mission). This new need also brings new challenges, such as workflow configuration and service placement.

One of the initial motivations of cloud computing is to reduce the burden of IT system management from end users and move this responsibility to cloud providers. As a result, end users can focus on their core business endeavors, while cloud providers offer easily manageable, highly elastic, and pay-as-you-go resources. However, without a reliable and adaptive (perhaps intelligent) cloud management system, a cloud is merely a cluster of servers sitting in a remote datacenter. As cloud computing becomes more popular, users are generating more complicated business processes or scientific jobs for the cloud.

Although biased, I like to think of these Web capabilities as being part of a workflow of services or service workflow. As developers begin to interleave Web capabilities across cloud infrastructures, deploying service workflows will introduce a number of challenges from a resource-allocation perspective. This leaves the following open questions.

When services in a workflow must be treated unequally during resource allocation, how should different services be ranked? When service capabilities are aggregated together into a service workflow, they usually have different priority levels to the workflow owner's business goal. For example, in an e-commerce workflow, the service that’s responsible for processing end users' orders is higher priority than the service that records users' browsing behaviors, because the former directly affects the workflow owner's revenue. Ideally, the cloud provider should identify those critical services in a workflow and give preferences to them during resource allocation and management processes.

How can different services be ranked without knowing the services' implementation details? Because a cloud platform must be capable of accommodating various kinds of workflows, and the cloud provider doesn’t have a priori knowledge of a workflow’s semantics, it's challenging for the cloud provider to rank different services while also treating them as black boxes.
What are the requirements for automatically determining the amount of resources allocated to a service? Usually a workflow has a user-specified budget limit as its constraint. The cloud provider must consider this constraint during the workflow configuration process. The challenge is to determine, without user intervention, the appropriate level of resources to allocate to each of the workflow’s component services. Furthermore, the budget is not the only constraint in the resource management process; the cloud provider also must consider the availability status of cloud resources to prevent resource overloading and contention.

What’s the most effective approach for managing the allocated resources of a workflow during its execution so that the workflow owner doesn’t need to do it manually? While executing a longstanding workflow, the amount of external input changing over time causes a fluctuation in the workflow’s workload level. In this case, the initial resource allocated to

Welcome Aboard!

Please join us in welcoming Yong Cui to the IEEE Internet Computing editorial board; he’s also the editor for the magazine’s “Standards” department.

Yong Cui is currently a full professor at the Computer Science Department in Tsinghua University. His major research interests include mobile cloud computing and network architecture. Cui has a PhD in computer science and engineering from Tsinghua University, China. He has served or served on the editorial boards of IEEE Transactions on Parallel and Distributed Systems and IEEE Transactions on Cloud Computing. He has co-authored seven Internet standard documents (RFC) for his proposal on IPv6 transition technologies, and he co-chairs the IETF IPv4/IPv6 Transition Softwires Working Group. Contact him at cuiyong@tsinghua.edu.cn.
From the Editors

Each service could become insufficient or redundant. To keep satisfactory service quality, or to save cost, it’s desirable for the resource management mechanism to proactively allocate more resources to services in need and release idle resources, avoiding unnecessary costs.

As you can see, resource allocation for cloud environments is an important challenge when managing the operations of a disparate group of Web capabilities. New infrastructures will be a strong requirement to manage an array of capabilities while customizing computing capabilities to the nature of the underlying jobs. Several of the articles in this special issue on “Cloud Storage” consider information management in the context of efficiently operating cloud environments.

The ideas I presented here (regarding resource allocation in cloud environments) were facilitated by discussions with David Wei of Microsoft. I would like to thank this issue’s guest editors, Robin Chen and Fred Douglis, for their excellent work. Moreover, I thank the authors for their articles, the reviewers for their service, and our readers for their interest. I hope that you enjoy the issue.

M. Brian Blake is the provost and executive vice president of academic affairs at Drexel University. As a professor of computer science and electrical engineering, his research interests are in service-oriented computing, adaptive distributed systems, and Web-based software engineering. Blake has a PhD in information and software engineering from George Mason University. Contact him at mbrian.blake@drexel.edu.

Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.