The Internet is populated with a plethora of different applications and services, some of which attract millions of users on a daily basis. E-business, online auctions, search, streaming, online social networking, and content sharing are just a few examples of currently popular applications that users access from a multitude of devices, ranging from smartphones to laptops to desktops. But with the increasing demand for mobility support and user customization as well as the growing complexity and richness of application interfaces and features, the workloads imposed by users on such applications exhibit nontrivial, unique, and hard-to-predict properties. We can define a system’s workload as a set of all inputs that the system receives from its environment, during any given period of time. For instance, if the system under study is a search engine, then its workload consists of all queries processed by it during an observation interval.

A solid understanding of how users typically behave when interacting with an application and the workload patterns derived from such behavior help us identify not only users’ needs but also application features that are more useful and attractive as well as possible system vulnerabilities. Moreover, the performance optimization, tuning, and management of a system with many clients and complex server and network infrastructures, which is typical of many popular Internet applications, depend heavily on an accurate knowledge of the workload typically experienced by it.

Thus, the measurement, characterization, and modeling of real workloads are key steps driving the design of new cost-effective applications and services as well as the optimization of existing ones. Workload characterization consists of identifying the basic components that compose the target workload, which depends both on the nature of the target application and on the purpose of the characterization. Workload modeling consists of building a representation that mimics the real workload under study, based on the identified components, whereas workload measurement, a key step to
all tasks in performance engineering, relates to gathering representative datasets to support the characterization task, helping us obtain workload parameters and establish a link between the real workload and its model.

Workload features have significantly contributed to the design and construction of many efficient systems. A classical example is the study of Internet traffic patterns that led to the development of efficient caching mechanisms and the deployment of effective content distribution networks. The measurement and characterization of a real workload can reveal workload peculiarities and invariants, helping system administrators size and manage their infrastructures. They might also reveal unwanted or unexpected patterns caused by various types of malware (such as viruses) or non-cooperative user behavior (such as spamming) and can support the design of effective detection and fighting mechanisms. Realistic workload models, in turn, can support cost-effective capacity sizing and management decisions, and drive performance analysis and optimization studies. They can also assist in generating synthetic workloads for experimental purposes.

Challenges

However, some of the most fundamental concepts and methods related to the characteristics of real Internet workloads are largely unknown to most Internet and Web practitioners. Consequently, many Internet-based systems and services are designed and built without exploring the inherent properties of their workloads, thus potentially producing suboptimal, if not poor, performance.

One possible reason is that workload characterization and modeling can explore a variety of different techniques, ranging from standard inferential statistics to more sophisticated methods including Markov models, clustering, principal-component analysis, and other data mining techniques, to name a few. Unfortunately, these techniques aren’t common knowledge to many system and performance engineers, a problem that’s further aggravated by the limited availability of real representative workloads for analysis, either due to privacy restrictions imposed by service providers or to the inherent limitations of partial/local workloads in the face of the global nature of Internet applications. Indeed, measurement-based studies often rely on data sampling, thus raising the issue of a possible sampling bias and its implications for accurate workload characterization.

Even if based on sampling, workload studies often need to capture, store, and process very large datasets, thus requiring an infrastructure with high processing, storage, and bandwidth capacities. Moreover, gathering and processing all that data can take a lot of time. Workload characterization and modeling involve a lot of analytical thinking, which can also take a lot of time, effort that some system administrators and engineers might not be able to undertake before making their decisions.

One major challenge to any workload study is the identification of qualitative patterns that hold across different workloads of the same target application, and, if possible, of different applications of the same type (search engines, video streaming applications), despite possible quantitative differences and idiosyncrasies. Such invariant patterns represent strong evidence of the expected workload, providing more valuable and accurate insights into application design, optimization, and management. Invariants can also be exploited for detecting anomaly and misuse, an increasing concern of any Internet system administrator. However, identifying such patterns relies on the availability of multiple datasets, collected from different applications, across different periods of time. Once again, data availability is often a restricting factor.

Another important issue concerns the temporal evolution of the analyzed workload. Internet workloads typically exhibit temporal variations, with very distinct daily and weekly patterns. If we analyze the workload over a period of great variability, treating it as a static snapshot, the analysis will undoubtedly reflect an “average” behavior, which might not accurately describe the workload experienced by the
system at any time interval. On the contrary, a sound workload characterization should be performed over periods of approximate stability, to avoid introducing spurious effects due to the aggregation of multiple workloads.

Another related concern has to do with characterizing and modeling evolutionary patterns. Internet applications and services are constantly and quickly evolving, as are the workload patterns they experience. Take, for instance, online social networking applications, such as YouTube, Facebook, and Twitter. As Walter Willinger and colleagues argued, most previous analyses of such applications, particularly of YouTube, have treated them as static, focusing on characterizing structural properties of a single snapshot of the relationship networks (such as friendship networks) built in to such systems. However, because online social networks are inherently dynamic, these studies fail to address key properties of the underlying system dynamics and, thus, of its workload. A clearer understanding of some of these dynamic properties could be exploited in the design of several mechanisms to optimize system performance. As an example, understanding content popularity evolution can drive the design of popularity prediction techniques, which in turn, could support the design of more cost-effective caching and content delivery strategies. Privacy concerns are another major challenge for collecting online social network data — for example, several successful attacks to anonymized workloads have been able to identify individual users.

In This Issue

This special issue features three articles that characterize and model workloads of different types, covering currently popular applications and drawing useful insights for system planning, sizing, management, and optimization.

In the first article, “Grid Computing Workloads,” Alexandru Iosup and Dick Epema present an extensive characterization of grid workloads. They analyze 17 traces from various commercial and research grids around the globe, including workloads that span the past seven years. They performed their characterization along four axes, namely, system usage (utilization and task submission patterns), user population, resource (CPU, memory, disk, and network) usage, and grid-specific application types. Regarding grid application types, the authors characterize the presence of bags of tasks, workflows, and pilot jobs. Many of their findings, which hold across several of the analyzed datasets, should be of great use to help us understand how grids and clusters are used, which in turn, can support the future design and tuning of new and existing grid systems.

In the article “To Cache or Not to Cache: The 3G Case,” Jeffrey Erman and colleagues found that 82 percent of the average downstream traffic on a large 3G wireless network in North America consists of HTTP traffic. This finding raises the question of the potential benefit of HTTP forward caching in 3G networks. The authors search for evidence of such potential by analyzing a dataset containing billions of HTTP requests, collected from a large 3G network. They characterize object popularity distribution and quantify the cache hit ratio across all requests, analyzing how it increases with consumer population. Their findings show great similarities with corresponding results in wired networks, where forward caching has been quite successful for many years. The authors also provide a caching cost model that can be used to determine the cost-benefit trade-offs of deploying forward caching at different levels in a 3G network hierarchy.

Finally, in the article “Analyzing and Modeling Workload Characteristics in a Multiservice IP Network,” Yongmin Choi, John A. Silvester, and Hyun-chul Kim characterize the workload of multimedia services in a nation-wide commercial network in Korea. By surveying the evolution of broadband Internet service in the KT Corporation’s network, the authors identify an increase in traffic from recently introduced IPTV and VoIP services. Focusing on the video on demand (VOD) component of IPTV services as well as on the VoIP service, they analyze their workloads at the session level, characterizing and modeling VOD content lengths as well as VOD and VoIP request arrivals and session durations. The characterization of workloads in a real large-scale production environment is quite difficult, as public data is rarely available. As the authors mention, the findings on the workload characteristics of popular IP multimedia applications provide valuable insights for network capacity planning and other network engineering tasks.
The articles in this special issue are examples of workload analyses that bring relevant insights for system research, design, and development. They contribute to our understanding of the characteristics of modern Internet workloads. But they also provide helpful information for constructing workload models that researchers can use to study the behavior of complex systems in controlled environments.

Acknowledgments

We thank the authors of all submitted articles and the reviewers for the constructive comments and help in the selection process. We also express our gratitude to Michael Rabinovich, IC’s editor in chief, Fred Dougis, the magazine’s editor in chief emeritus, Maarten van Steen, its associate editor in chief, and the production staff at the IEEE Computer Society. Their prompt help and hard work made this issue possible.

References


Virgilio A.F. Almeida is a professor of computer science at the Federal University of Minas Gerais, Brazil. He has a PhD in computer science from Vanderbilt University. Almeida is a recipient of a Fulbright Research Scholar Award. Contact him at virgilio@dcc.ufmg.br.

Jussara M. Almeida is an associate professor of computer science at the Federal University of Minas Gerais, Brazil. Her research interests are primarily in performance modeling of large-scale distributed systems. Almeida has a PhD in computer science from the University of Wisconsin-Madison. Contact her at jussara@dcc.ufmg.br.

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