Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Synthetic workload generation for capacity planning of virtual server environments

Hugo E. S. Galindo, Wagner M. Santos,
Paulo R. M. Maciel, Bruno Silva and Sérgio M. L. Galdino
Centro de Informática
Universidade Federal de Pernambuco
Recife, Pernambuco, Brasil

José Paulo Pires
Hewlett-Packard (HP) Brasil
Porto Alegre, Rio Grande do Sul, Brasil

***Abstract*—Capacity planning tools aim at monitoring and evaluating computational infrastructures resources in terms of workloads. The *Capacity Advisor* is a capacity planning tool inside the *HP Virtual Server Environment - VSE*. This paper describes the development and use of a synthetic workload generator named *WGCap* (*W*orkload *G*enerator for *Cap*acity Advisor), a flexible synthetic workload generator that generates traces to be imported into the *Capacity Advisor* for simulating the consumption of resources like CPU demand, memory size, disk data rate and network data rate. The workload generation is reliable and flexible, allowing the generation of workloads based on actual traces or on a significant set of probability distributions and statistical summaries.**

***Index Terms*—Synthetic workload generation, capacity planning, random variates.**

I. Introduction

Information systems are increasingly present in all activities of our daily lives and in the most diverse areas of knowledge, from health, education and communication to finance, security and entertainment. Given the widespread use of information systems, performance of services provided have been object of attention of service providers, infrastructure managers, application designers and the scientific community.

Capacity planning consists on the definition of the necessary resources in a way that the services provided (accomplished) by the systems, meet the appropriate quality levels (response time, level of availability etc.) to the processes that use these services or resources, considering future demands. In fact, for the successful accomplishment of such an activity, the resources should be designed and tuned to minimize the associated costs and to meet the established service level agreement.

Capacity planning tools aims at monitoring and evaluating computational infrastructure in terms of *workloads* (values removed from the tracking of real systems or produced synthetically) allowing the analysis of resource utilization and planning of system's infrastructure.

The performance evaluation of systems considering workloads generated from real traces is an important feature, but a workload generator should also generate workloads based on statistics figures, since the real traces might not be available to the analyst. The reasons why an organization may not make the traces available for analysis encompass confidentiality of content and the non-availability of trustful set of significant traces.

Many works related to synthetic workload generation have been conducted over the years in several areas of knowledge. [10, 12] have proposed approaches for validating synthetic disk request generators. [6] shows the design and use of a synthetic workload generation tool for simulation of web proxy caches, called ProWGen. [14] shows another tool called WebTraff. The tool is an evolution of ProWGen that has an interactive graphical user interface (GUI) that can be used for modeling and analysis of Web proxy workloads. [5] introduces SWORD (Scalable WORkloaD generator), a workload generator for testing and benchmarking of high-volume data processing systems.

This work concerns the development of a flexible mechanism for workload generation applied to a capacity planning tool for virtual environments, the *HP Capacity Advisor*. The synthetic workload simulates the consumption of main computational resources (CPU demand, memory size, disk data rate and network data rate) and it is generated in a trace format compatible with the Capacity Advisor, so the traces generated can be imported into the tool easily and quickly.

This research has two main objectives: development of methodology and computational prototype for supporting workload generation based on actual traces; development of methodology and computational prototype to support the generation of workload based on a significant set of probability distributions and statistics summary.

The paper is organized as follows. Section II details the Capacity Advisor (capacity planning tool of the *HP Virtual Server Environment - VSE*), providing an overview of its operations, architecture and trace format. Section III shows the WGCap workload generation approach proposed, the main scenarios and mechanisms adopted. The section also presents the creation and validation of the synthetic traces. Case studies are shown in Section IV, where the synthetic traces generated by the WGCap are imoported into Capacity Advisor. Finally Section V concludes the paper and presents ongoing and future works.

II. Capacity Advisor

This section introduces the environment where the workload generator is applied, where the VSE and the main features of

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

Capacity Advisor and its architecture are depicted.

A. VSE - Virtual Server Environment

The *HP Virtual Server Environment (VSE)* [4] is a virtualization solution developed by *HP (Hewlett Packard)* that provides an administration and monitoring environment of virtual servers and their applications. The VSE works as an integrated set of multiplatform products and technologies to help maximizing the use of server resources. *The HP VSE Management Software* provides visualization, configuration, workload policy, application management, and capacity planning tools to optimize system resources. The software provides an integrated graphical environment, through *HP SIM*[1], for managing physical servers, logical servers, virtual machines, server blades, nPartitions, virtual partitions, applications and workloads. [3, 2].

The HP VSE provides control of virtualized environment through an integrated software suite for planning, management and automation, that includes: *HP Virtualization Manager*, *HP Global Workload Manager (gWLM)*, ***HP Capacity Advisor***, *HP Integrity Virtual Machines Manager (VM Manager)*, *HP Application Discovery*, *WBEM providers* and *other VSE agents*.

B. The Capacity Advisor: an overview

The Capacity Advisor is a component of the VSE Management Software, acting as a capacity planning tool for the environment. It enables capacity planners to analyze historical *workload*[2] utilization data for planning workload migrations or new workload undertaking. It automates the steps that are traditionally performed manually by capacity planners, thereby freeing the capacity planners to focus on planning for future workload introductions, migrations, and consolidations.

The Capacity Advisor tool may collect workload utilization data for every workload in the enterprise data center. This data is collected on a daily basis from each system and is stored on the HP Systems Insight Manager CMS (Central Management Server).

C. Capacity Advisor Architecture

The Figure 1 shows the main components that work with the Capacity Advisor and the architecture of its operation.

The access to Capacity Advisor is done through a web browser on any machine configured to access the Central Management Server (CMS), a system in the management domain that executes the HP SIM. All central operations within HP SIM are initiated from this system. The web interface is provided by HP SIM, which also functions as an interface to access the VSE Management Software. The Capacity Advisor is a component of the VSE Management Software and its operation will depend on the correct installation and configuration of HP SIM and the VSE in the CMS.

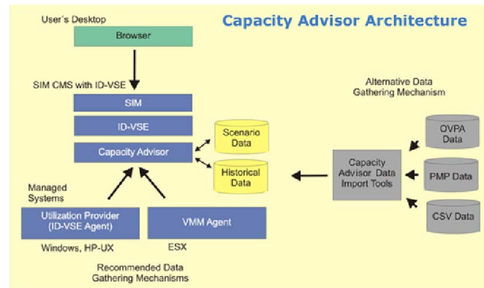

Fig. 1. Capacity Advisor Architecture

There are two types of data gathering mechanisms for Capacity Advisor, the recommended mechanisms and the alternative mechanisms [1].

Recommended data gathering mechanisms: are made by agents at the systems to collect historical data on resouce utilization. On systems managed by the VSE (Windows and HP-UX) is used the Utilization Provider and on ESX systems is used the VMM (Virtual Machine Manager) Agent. Traces of historic data are gathered for CPU utilization, memory utilization, disk I/O capacity utilization and network I/O capacity utilization. Data values are collected at 5-minute intervals and are stored in VSE database. Then the Capacity Advisor Data Collector gathers data from the VSE database and includes it in the Capacity Advisor Database (Historical Data).

Alternative data gathering mechanisms: Another option is to import the data directly to the Capacity Advisor Database. Importing data can be made using OVPA Data (gathered from HP OpenView Performance Agent), PMP Data (gathered from HP Performance Management Pack) and CSV Files (files in CSV format exported from Capacity Advisor). The Capacity Advisor Data import Tools convert the data automatically and import it into the Capacity Advisor database.

D. Trace Format - CSV files

The Capacity Advisor can import and export data collected from workloads or systems in traces structured in CSV (Comma-Separated Values) file format. The data is imported and exported as a series of lines containing the following:

- profile headers, including date and metric labels and
- utilization values.

The collection period is of 5 minutes. Each data line is called a sample. The first eight lines of the import file are format headers that define the profile characteristics. This information is used to normalize the utilization data when it is saved in the Capacity Advisor database. The information includes a *profile name, hostname, CPU count and speed, memory size, OS platform or type, model information, and product version number*. These are followed by *date* and *metric labels* [1].

[1]*HP Systems Insight Manager*: a centralized management tool that provides consistent multisystem management capabilities. Virtualization Manager is tightly integrated with HP SIM; as a result, all of the features available from HP SIM are also readily available when using the VSE management suite. [13]

[2]*workload*: In VSE a *workload* is a collection of processes within an OS image running on a managed system whose performance is managed as a single unit. Examples include the processes that belong to an application or all processes owned by a specific user. [13]

The format header is structured as follows:

#Profile: *name*
#Host: *hostname*
#CPU: CPU_*count*@CPU_*speed*GHz
#Memory: MEM_*size*MB
#OS: *platform*
#Model: *model*
#Version: *version_number*
[YYYYMMDDhhmm,] UTIS, *metric* [, *metric*, ...]

The last header line contains the comma-separated list of labels for the date and gathered metrics, with or without spaces, where the fields are defined as:

YYYYMMDDhhmm - Timestamp in local time, given in units of YYYY (year), MM (month, as 01 to 12), DD (day, as 01 to 31), hh (hours, as 00 to 23), and mm (minutes, as 00 to 59).

UTIS - Universal Time (GMT) in seconds (standard UNIX time in seconds since 1 January 1970).

metric - At least ***one*** of the following: CPU_ALLOC, CPU_UTIL, DISK_UTIL, MEM_ALLOC, MEM_UTIL, NET_UTIL, PHYS_CPUS, PHYS_MEM.

III. Workload Generation

As noted earlier, the main objective of this work is the generation of workloads (synthetic traces) to be imported into Capacity Advisor. This section introduces the main mechanisms and strategies used to ensure that this generation can occur.

The Figure 2 illustrates the Capacity Advisor architecture shown in the previous section with the addition of the proposal for workload generation of this work.

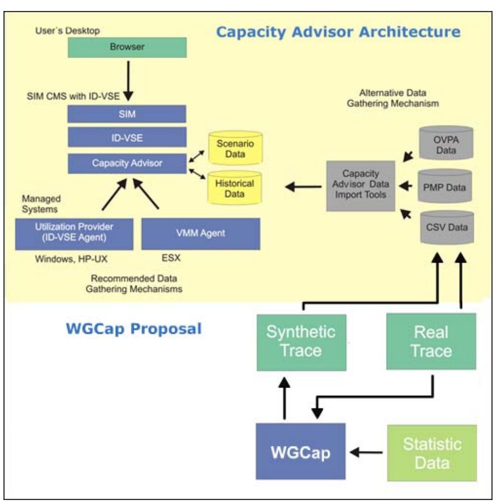

Fig. 2. Capacity Advisor Architecture and WGCap Proposal

The WGCap works with two types of input data for generating the synthetic traces: *real traces* (colected from actual systems) or *statistical data* (calculated from the selected computational resources). The output generated by WGCap is a CSV file format with the same structure of the files exported by Capacity Advisor with the values of each resource synthetically generated, so the synthetic traces generated can be imported into the tool easily and quickly.

The generation can be performed accordingly to three scenarios: *generating synthetic traces from real traces*, *generating synthetic traces from statistical summaries* and *generating statistical summaries and synthetic traces from real traces*. These scenarios will be described in detail in the next subsection.

A. WGCap Scenarios

Scenario 1 - This scenario is responsible for allowing user's system to generate synthetic traces from actual traces. Through real traces, the user can generate synthetic traces that follow the behavior of the real trace provided. For example, the user considers a trace of a week and can generate a significant trace for a year.

Scenario 2- The tool can generate synthetic traces receiving as input the statistical summaries provided by user. The representation of each resource in the trace (IO, disk, memory, CPU) is based on a probability distribution implemented in the tool. After calculating the statistical summaries, the user must choose a probability distribution to determine the behavior of each resource.

Scenario 3 - This scenario addresses the situation where the customer can not provide or arrange the real traces to be directly used for generation of synthetic traces, however the traces can be used for the generation of statistical summaries. In this scenario, the synthetic traces are generated from the statistical summary calculated from the real trace, working the same way as Scenario 2.

B. Methodology for generation of synthetic workloads

Regardless of input data, in each of the 3 scenarios, the methodology used for synthetic workload generation, which represent the values of each resource, are made through the strategy of generating *Random Variates* [9, 11].

Random variates are used when simulating processes driven by random influences (stochastic processes). In modern applications, such simulations would derive random variates corresponding to any given probability distribution from computer procedures designed to create random variates corresponding to a uniform distribution, where these procedures would actually provide values chosen from a uniform distribution of pseudorandom numbers.

Procedures to generate random variates corresponding to a given distribution are known as procedures for random variate generation [7]. The most popular methods of random variate generation, used in our research, are: *The inverse transform sampling method*, *Composition*, *Convolution* and *The Rejection sampling method*.

In our research, initially, a tool which applies these methodologies was developed. The tool will be described in detail in the next subsection.

C. The WGCap kernel

To perform the generation and validation of synthetic workloads, a tool that works as the kernel of the synthetic workload generator was initially developed. The main objective of this kernel is to generate samples of some probability distributions and validate the algorithms of generating random variates. The libraries created and used in the kernel are fundamentally important in the context of the project, taking a key role in the generation of workloads for the Capacity Advisor.

With the WGCap kernel is currently possible:

- To generate any quantity of random numbers based on 8 probability distributions (Exponential, Normal, Erlang, LogNormal, Geometric, Pareto, Weibull, Poisson). To generate the samples it is necessary to define the parameters values of each used distribution and set the number of bins (classes) for the histogram;
- To generate random numbers empirically (Empirical distribution). In this case, when selecting an empirical distribution, loading a file with a set of values that represent any behavior is required. From there, random numbers with the same behavior of the loaded file will be generated;
- View the histogram of the data generated, and may experience a graphic comparison with the original probability distribution;
- To view the statistical summary of the generated data;
- To perform the distribution fitting of the samples: Kolmogorov-Smirnov Goodness-of-fit test;
- To save the values of the samples and statistical summary generated, in text files for use later.

The Figure 3 illustrate the WGCap kernel GUI and its main features. In this example $100,000$ samples based on a normal distribution were generated.

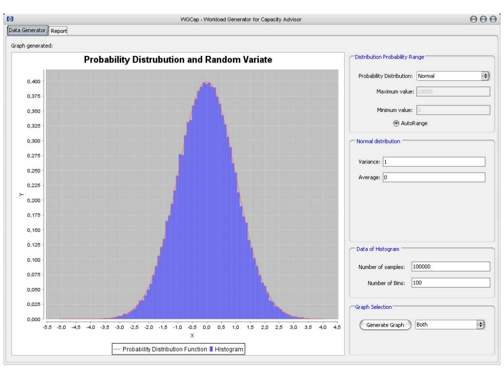

Fig. 3. WGCap Kernel GUI - Snapshot 1

To validate the samples generated by the kernel, besides the validation tests made in the tool, a tool of distribution fitting was used to prove that the values of the random numbers generated based on the probability distributions implemented, actually correspond to the probability distribution of origin.

$10,000$ samples from each distributions (continuous and discrete) were generated and goodness of fit tests for each sample generated were performed. The following tests were applied: Kolmogorov-Smirnov, Anderson Darling and Chi-Squared [8]. Figures 4, 5 and 6 shows the results achieved.

Kolmogorov-Smirnov

Distributions	Statistic	Alpha	Critical Value	Reject?
Erlang	0,00256	0,01000	0,00515	No
Exponential	0,00250	0,01000	0,00515	No
Geometric	0,00183	0,01000	0,17182	No
LogNormal	0,00167	0,01000	0,00515	No
Normal	0,00223	0,01000	0,00515	No
Pareto	0,00231	0,01000	0,00515	No
Poisson	0,00142	0,01000	0,21043	No
Weibull	0,00266	0,01000	0,00515	No

Fig. 4. Kolmogorov-Smirnov test results

Anderson-Darling

Distributions	Statistic	Alpha	Critical Value	Reject?
Erlang	1,11700	0,01000	3,90740	No
Exponential	0,82915	0,01000	3,90740	No
LogNormal	0,39719	0,01000	3,90740	No
Normal	0,37177	0,01000	3,90740	No
Pareto	2,67120	0,01000	3,90740	No
Weibull	0,81488	0,01000	3,90740	No

Fig. 5. Anderson-Darling test results

Chi - Squared

Distributions	Statistic	Alpha	Critical Value	Reject?
Erlang	22,03700	0,01000	32,00000	No
Exponential	16,41900	0,01000	32,00000	No
LogNormal	15,77000	0,01000	32,00000	No
Normal	9,15230	0,01000	32,00000	No
Pareto	12,42600	0,01000	3,90740	No
Weibull	18,69500	0,01000	3,90740	No

Fig. 6. Chi - Squared test results

Note that all samples based on probability distributions generated by the WGCap kernel were accepted by at least one of the tests applied.

To validate the empirical distribution, $8,928$ samples of CPU utilization collected from a real trace representing a particular behavior were imported into the tool. Then the same amount of samples was generated through the empirical distribution. The result of the comparison between the two samples (real and empirical) is shown in Figure 7. It can be seen that the two samples are equivalent.

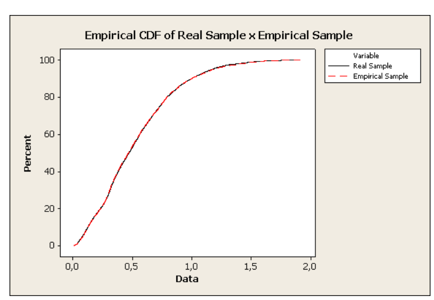

Fig. 7. Empirical CDF of Real Sample x Empirical Sample

D. Generating a trace

After validating the values generated by the tool, we can finally generate the traces in CSV format to be imported into Capacity Advisor. The steps for the trace generation are described below.

1) Define the characteristics of the environment to be simulated. These information must be included in the header of the trace:
   #Profile: *name*
   #Host: *hostname*
   #CPU: CPU_*count*@CPU_*speed*GHz
   #Memory: MEM_*size*MB
   #OS: *platform*
   #Model: *model*
   #Version: *version_number*
   [YYYYMMDDhhmm,] UTIS, *metric* [, *metric*, ...]
2) Define the *timestamp* (starting date) for the simulation of the traces. The date should be in the following format: *YYYYMMDDhhmm*. The remaining dates will be created automatically according to the number of samples to be generated.
3) Select the metrics of the resources that will be used. At least one of the following:
   CPU_UTIL, DISK_UTIL, MEM_UTIL, NET_UTIL.
4) Select the probability distributions that will be used to generate the values of each metric chosen. In this step must be defined the values of each parameter of the distributions, these parameters may be defined by the statistic data provided by user (Scenario 2) or generated by the tool (Scenario 3). In empirical distribution, used to generate synthetic traces from actual traces (Scenario 1), it is required to load a file with a set of values that represent any behavior. From there, random numbers will be generated with the same behavior of the loaded file.
5) Define the number of samples and bins (classes for the histogram) to be generated.
6) Generate the trace.

After these steps, the trace can be finally imported by the Capacity Advisor.

IV. Case Studies

This session presents two case studies involving the scenarios shown in the previous section. Synthetic traces, based on actual traces and statistical summaries, were generated for simulating the consumption of all resources available at Capacity Advisor (CPU demand, memory size, disk data rate and network data rate). After generation, the synthetic traces were imported into Capacity Advisor.

A. Generation of a synthetic trace based on a real trace with the same number of samples

This case study shows the creation of a synthetic trace with samples of the metric DISK_UTIL (disk I/O capacity utilization) based on samples from a real trace generated by Capcity Advisor.

The real trace characteristics were defined by the following profile header:

#Profile: legacy11
#Host: legacy11
#CPU: 2 @ 2.7GHz
#Memory: 2 GB
#OS: WINNT
#Model: ProLiant DL140 G3

In this profile 8,928 samples were collected from each resources used in a system in production. The initial timestamp, which refers to the first sample of the trace, was 200708311800 (YYYYMMDDhhmm). The subsequent samples were collected in 5-minute intervals (increments of 300 seconds).

To generate the synthetic trace, initially it was created a new profile header with the same settings of legacy11 profile (CPU, Memory, OS and Model) and the same initial timestamp (200708311800). Then the metrics of the resources that should be synthetically generated were selected, in this case we selected the metric DISK_UTIL. After, an empirical distribution based on the samples of DISK_UTIL, taken from legacy11 profile, was generated. Finally the number of samples to be generated by the tool was selected, as the objective was to generate a synthetic trace with the same number of samples of the real trace, we selected 8,928 samples.

After generating the values, the synthetic trace was imported to the Capacity Advisor. The Figure 8 illustrates the achieved results, representing the disk usage in the period from 8/31/2007 to 10/3/2007.

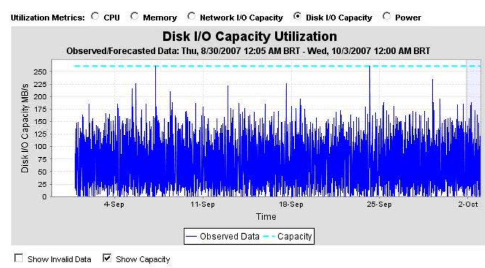

Fig. 8. Disk I/O Capacity Utilization

B. Generation of a synthetic trace based on probability distributions and statistical summaries

This case study shows the creation of a synthetic trace with samples of the metric MEM_UTIL (memory utilization) based on a probability distribution and statistical summary. The main aim of this case study is to generate a synthetic trace even when a real trace is not available, by using statistics and a set of probability distributions to define the behavior of the workloads to be generated.

The statistics used for generating the synthetic trace were extracted from the legacy11 profile, and the trace to be simulated will also follow the same definitions of the legacy11. Thus, the synthetic trace characteristics were defined by the following profile header:

#Profile: Synthetic_legacy11
#Host: Synthetic_legacy11
#CPU: 2 @ 2.7GHz
#Memory: 2 GB

#OS: WINNT
#Model: ProLiant DL140 G3

After creating the new profile header, we defined the initial timestamp. As this measure does not affect the generation of values, we decided to use the same initial timestamp defined in legacy11 - 200708311800. Then, we selected the metrics of the resources that should be synthetically generated, in this case the metric MEM_UTIL was selected. After, we selected the probability distribution that is used to generate the values of the metric chosen, at this stage we chose the Normal distribution.

The legacy11 statistics were extracted by WGCap and the parameters were inserted in the distribution as follows:

MEM_UTIL
Distribution: *Normal*
Parameters: Average = $1,2005$ / Variance = $0,0349$

Finally, we selected the number of samples to be generated by the tool, in this case we chose $10,000$ samples.

After generating the values, the synthetic traces were imported to the Capacity Advisor. The Figure 9 illustrates the achieved result, representing the consumption of disk in the period from 8/31/2007 to 10/3/2007.

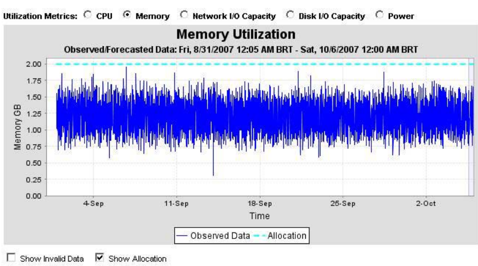

Fig. 9. Memory Utilization with Normal distribution

These case studies showed the results of importing the traces generated by WGCap to the Capacity Advisor. The use of the developed tool is extremely important for the capacity planning and tuning of virtual servers, being able to manipulate traces collected from an existing infrastructure or work with statistical data, simulating the behavior of an inexistent infrastructure. Thus, the planning of infrastructures performed by capacity planners can be done in a more flexible way.

V. Conclusions

This paper presented the WGCap, a flexible synthetic workload generator for the HP Capacity Advisor. The objective of the tool is to generate traces to be imported into the *Capacity Advisor* for simulating the consumption of resources like CPU demand, memory size, disk data rate and network data rate. The workload generation has been successfully validated and shown to be reliable and flexible, allowing the generation of workloads based on actual traces or on a significant set of probability distributions and statistical summaries.

Two case studies were developed and the synthetically generated traces were imported to the Capacity Advisor successfully and the results were seen in a satisfactory manner.

In future studies, we intend to implement a new set of probability distributions, with the aim of increasing the number of possibilities of the tool. We also intend to build a more complete and integrated tooling, making the workload generation more intuitive and fully automatic, since some steps performed to generate the synthetic workloads are still carried out manually.

References

[1] HP Capacity Advisor Version 4.1 User's Guide, January 2009.

[2] HP Insight Dynamics - VSE and HP VSE Management Software 4.1 Getting Started Guide, January 2009.

[3] Implementing a Virtual Server Environment: Getting Started, January 2009.

[4] Introduction to the HP Virtual Server Environment, January 2009.

[5] K. S. Anderson, J. P. Bigus, E. Bouillet, P. Dube, N. Halim, Z. Liu, and D. E. Pendarakis. SWORD: scalable and flexible workload generator for distributed data processing systems. In L. F. Perrone, B. Lawson, J. Liu, and F. P. Wieland, editors, *Winter Simulation Conference*, pages 2109–2116. WSC, 2006.

[6] M. Busari and C. Williamson. ProWGen: A Synthetic Workload Generation Tool for Simulation Evaluation of Web Proxy Caches. In *Computer Networks*, volume 38, pages 779–794, June 2002.

[7] L. S. Cassandras, Christos G. *Introduction to Discrete Event Systems*. Springer,2nd ed., 2008.

[8] C. A. Chung. *Simulation Modelling Handbook: A Practical Approach*. Routledge, USA, 2003.

[9] L. Devroye. *Non-Uniform Random Variate Generation*. Springer-Verlag, New York NY, 1986.

[10] G. R. Ganger. Generating representative synthetic workloads: An unsolved problem. In *International Conference on Management and Performance Evaluation of Computer Systems (Nashville, TN*, pages 1263–1269, 1995.

[11] J. E. Gentle. *Random Number Generation and Monte Carlo Methods*. Springer, New York NY, 1998.

[12] M. E. Gómez and V. Santonja. A New Approach in the Modeling and Generation of Synthetic Disk Workload. In *MASCOTS '00: Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, page 199, Washington, DC, USA, 2000. IEEE Computer Society.

[13] B. Herington, D. & Jacquot. *The HP Virtual Server Environment: Making the Adaptive Enterprise Vision a Reality in your Data Center*. Prentice Hall, 2006. Cap. 17 & 18.

[14] N. Markatchev and C. Williamson. Webtraff: A GUI for web proxy cache workload modeling and analysis. In *In Proceedings of the 10th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS '02*, pages 356–363, 2002.