Short-term Performance Management by Priority-based Queueing

Christian Markl, Oliver Hühn, Martin Bichler

Department of Informatics, Boltzmannstr. 3

Technische Universität München

85748 Garching, Germany

{markl, oliver.huehn, martin.bichler}@mytum.de

Phone: +49 89 28917528

Fax: +49 89 28917535

Abstract

Service-based IT infrastructures serve many different business processes on a shared infrastructure in parallel. The automated request execution on the interconnected software components, hosted on heterogeneous hardware resources, is typically orchestrated by distributed transaction processing (DTP) systems. While pre-defined quality-of-service metrics must be met, IT providers have to deal with short-term demand fluctuations. Adaptive prioritization is a way to react to short-term demand variances. Performance modelling can be applied to predict the impact of prioritization on the overall performance of the system. In this paper we describe the workload characteristics and particularities of two real-world DTP systems and evaluate the effects of prioritization regarding overall load and end-to-end performance measures.

Keywords: Performance Modelling – IT Service Management – Transaction Processing - Prioritization - Capacity Management

1. Introduction

Today's services industries make use of distributed transaction processing (DTP) systems to run their day-to-day business. DTP systems coordinate the automated execution of business processes on shared IT infrastructures, often incorporating hundreds of basic service components hosted on numerous heterogeneous hardware resources. The automated request execution on such business processes without human interaction (referred to as *workflows* in the following) is typically orchestrated by a transaction processing (TP) monitor [1-3].

Capacity planning and performance management issues of such applications are business-critical for IT service providers as the DTP systems have to satisfy quality-of-service goals pre-defined in service level agreements (SLAs). Important performance measures include end-to-end workflow response times, throughput, and availability. While proactive planning of such a meshwork of IT components is already difficult, the demand variations in dynamic business environments make it even more challenging. The request ratios on the single workflows that are served in parallel (often referred to as *workflow mix*) as well as the overall demand of the system in a certain time period change over the year, the week, and also throughout the day.

In our research, we focus on the performance and capacity planning for DTP systems. We analyzed two systems of our industry partner, a telecommunication provider. The workload is stochastic: tariff changes as well as new products, marketing campaigns, activities of competitors, and seasonality influence the number of requests that has to be served. For example the number of new customer activations strongly increases in the weeks before Christmas. In addition to this user-initiated Online Transaction Processing (OLTP) workload, the DTP systems have to serve maintenance and management requests created by internal IT components such as the CRM system, billing, or backup strategies. These requests are typically fed into the system as batch jobs during the night as the OLTP workload is relatively low during this time period.

While many seasonalities and daily cycles in the workload can be forecasted with traditional time series methods, some short-term variations due to marketing campaigns or unforeseen moves of the competition are hard to predict based on historical data. Often, IT service providers deal with this problem by over-provisioning their system capacities, allowing for short demand peaks. As a consequence, the resulting resource utilization is quite low, as low as 10% in our systems under study.

An alternative way of dealing with short-term demand variances is adaptive workflow prioritization at runtime. By giving those workflows with unforeseen demand peaks a higher priority, costly capacity buffers can be reduced as performance measures of these workflows will be improved. However, of course, the effect of prioritization has an impact on the other workflows on the shared IT infrastructure. It is therefore important for the IT service providers to understand the impact of possible prioritization scenarios on the performance measures of the overall system. As benchmarking and testing of such complex IT infrastructures is often very costly and time-consuming, performance modelling can be used to estimate the impact.

In the remaining of this paper we first describe the particularities of the two real-world DTP systems of our industry partner. In chapter three, we then characterize the workload specifics of these systems in detail. After an overview of performance modelling in this area in chapter four, we evaluate the effects of

prioritization strategies on the overall system performance in chapter five. Related literature is discussed in chapter six. In parts, we draw on and extend previous work in [4]. The main focus of the extension lies in the revision of the prioritization levels and their influence on the overall system performance.

2. Overview of DTP Systems

Modern enterprise IT infrastructures consist of a wide variety of applications and systems. A DTP system supports the flexible composition of distributed software services in such heterogeneous environments to workflows, and provides an implementation of automated business processes.

Typically, several workflows are hosted on a shared IT environment, using the basic IT service components collectively. For example the validation of a credit card account might be such a basic service component that can be incorporated in several workflows of a shop system. Basic IT components might call others to fulfill their functionality, such as queries to a CRM or billing system. These interwoven call structures lead to a complex meshwork of interconnected IT services.

In practice, most automated business processes are executed on TP monitors such as Oracle Tuxedo [1], the Vitria suite [2], TIBCO Business Works [3], and a growing number of BPEL engines. The main purpose of a TP monitor is to ensure the ACID properties (atomicity, consistency, isolation, and durability) of the single transaction steps of a workflow while managing the automatic request execution across the shared IT infrastructure [5]. Coordinating this process across numerous services, running on many different software components, hosted on heterogeneous hardware resources, and connected by possibly unreliable communication links, is an essential problem in distributed transaction management. TP monitors face this challenge by rollback mechanisms and the two-phase commit protocol [6].

Technically, TP monitors are often implemented as a Message Oriented Middleware (MOM) to organize the eradication of all jobs. Figure 1 shows a part of a sample MOM. Service components send messages to the MOM; the MOM itself sends messages to the affected service components. The communication between service components contains information about the start of a new job or the termination a job. This information is saved in the message queues. If a service component sends a message that it can process a new job, the MOM sends the next job to the service component. This job can be the first job in the queue, based on the dispatching routine of the MOM. DTP systems also provide other queueing and dispatching disciplines such as last-in-first-out, or highest-priority-first. The latter requires the system administrator to define priorities for different types of workflow requests.

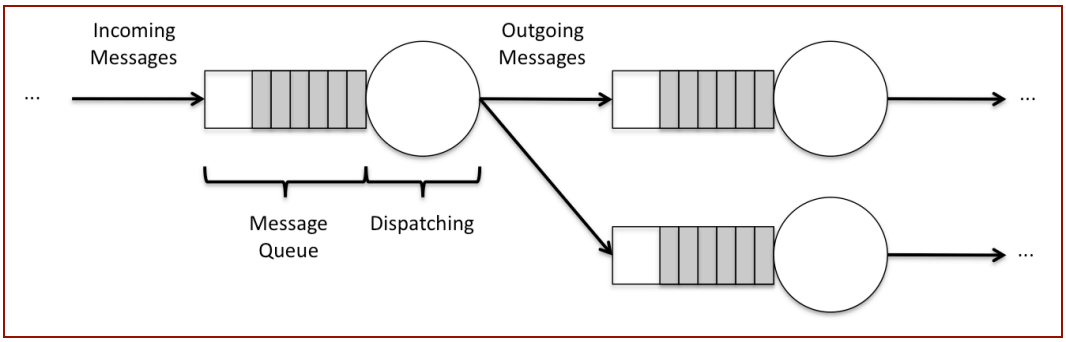

Figure 1: Structure of a message-oriented middleware

3. Workload of DTP Systems

Many enterprise systems, which serve a national market (in a single time zone), exhibit a characteristic workload pattern, with an increasing number of requests in the morning and a decline in the evening. While there are yearly seasonalities for customer-centric applications, which are bound to the sales cycles of a business, much of the variation can be explained by daily cycles.
Figure 2 shows the observed workflow workload pattern of an industry partner. The daytime consists of an OLTP workload initiated by human interaction on a web portal; its amount and arrival time can only be estimated but not be directly controlled by the IT service provider. During the night, several internal systems such as the billing or the CRM system run batch jobs on the DTP systems.
Daytime end-to-end response times of most workflows are the primary metric in SLAs as they will be observed by the customers. We will therefore focus on the daytime workload for our studies.

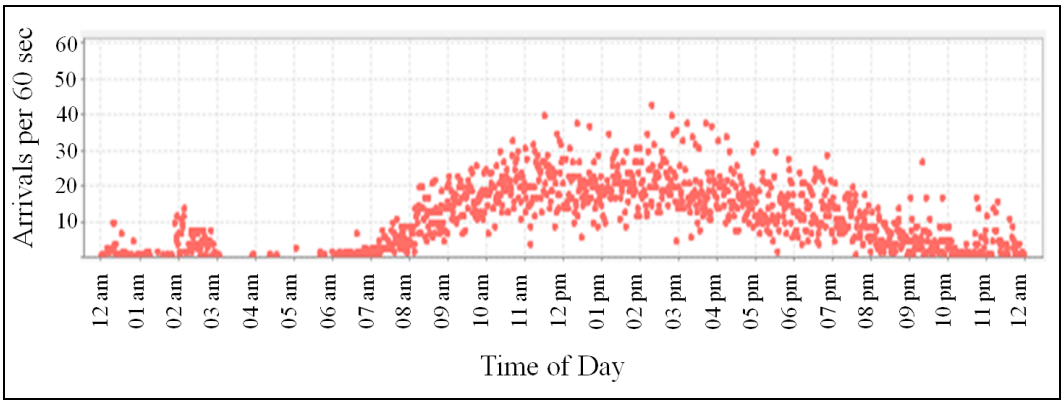

Figure 2: Typical workload scenario on a sample workflow of the DTP system of a telecom provider during the day

The workload of the workflow in Figure 2 is representative for most of the other workflows, which we have analyzed as well, however individual workflows have varying peak demand times. The result is a dynamically changing workflow mix throughout the day.
It is important to understand the expected overall load and the workload mix of the DTP systems for a planning period to meet adequate capacity planning decisions.
We applied the traditional time series analysis such as linear regression [7] or triple exponential smoothing [8] to extract seasonal patterns in the workload. While such forecasts represent most of the variation throughout the day, certain

days exhibit outliers or short-term demand peaks due to some unforeseen events. Such events could be due to successful marketing campaigns or product failures, which lead to a larger volume of complaints and other types of customer interaction with the system.
These short-term demand peaks cannot be predicted based on historical data and are one of the reasons for the over-provisioning of DTP systems, because the decision makers want to avoid the risk of violating pre-defined SLAs. As peak demands on single workflows are typically short and only impact one or a few workflows, adaptive prioritization is one possibility to address these short-term demand variances.
The workflows of the systems under study include different pre-defined SLAs regarding response time. The ones with direct customer interaction (like activation of a new mobile) are more business critical than other workflows. The goal of prioritization in our study is to guarantee the pre-defined SLAs for the business critical workflows even in times with short-term demand-peaks without strong over-provisioning.
Prioritization of workflows in such complex systems has side effects to overall system performance, and will impact the performance of other workflows. If the operator decides to adapt the priority of certain workflows in such situations, it is important to understand the impact of such changes on the overall system. Discrete event simulation and queueing networks are two methods to estimate and predict this impact.

4. Performance Modelling

Modelling real-world systems typically has two major goals: gaining insight into the actual system and predicting future system behavior [9]. Moreover, performance models support decision makers in their planning and optimization tasks by identifying possible bottleneck hardware components (referred to as *service components* in the following) or predicting the expected workflow response times of a workload scenario.

4.1. Overview

We apply Queueing Theory (QT) to model DTP systems. QT is a well-studied methodology for the analysis of systems with service stations and waiting lines. Its applications range from manufacturing system planning over computer processor design to multi-tiered web applications [9, 10].
The complex interdependencies of service components in modern DTP systems result in end-to-end workflow response times that develop in a non-linear way up from a certain amount of load. The strength of QT is, that once a valid performance model is built, this non-linear response time behavior can be predicted. Thus, it is possible to evaluate the impacts of many different load scenarios to the overall performance of the systems.
Queueing Network Models (QNMs) represent a system as a network of service stations with queues that serve requests of several classes. Figure 3 shows one exemplary QNM with three single service components. A single service component consists of one or more identical parallel servers with a joint waiting room, the queue. Jobs arrive at the queue with an arrival rate λ and have an expected service time E(S). Both the arrival rate and expected service time are modelled via distributions - in our case the exponential distribution. If the servers

of a service component are all occupied, jobs have to line up in the queue and wait for their execution.

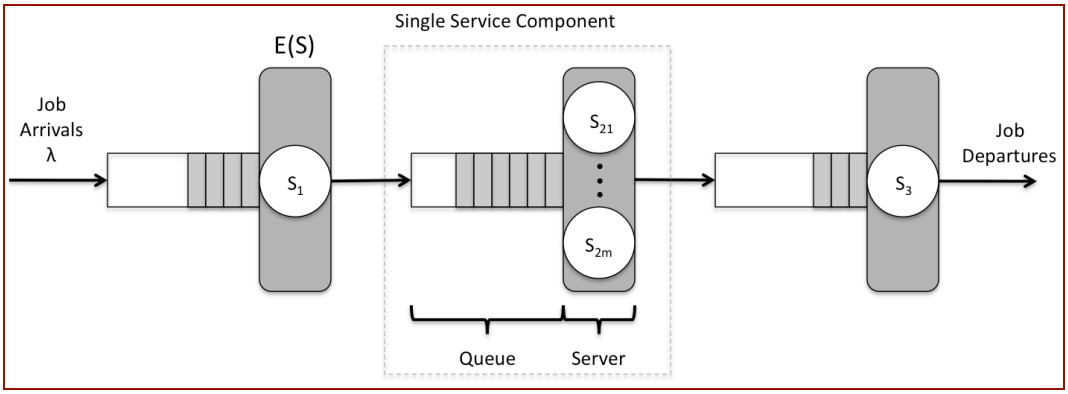

Figure 3: Queueing Network Model

Three types of queueing networks are distinguished:

- Open Queueing Networks,
- Closed Queueing Networks, and
- Mixed Queueing Networks

When the jobs enter the system from outside and leave it after being served, we speak of an Open Queueing Network Model. In Closed Queueing Networks, the jobs circulate inside the system without leaving it. If jobs of both characteristics are combined in a model, it is called a Mixed Qeueuing Network Model. DTP systems can be modelled as Mixed QNMs.
One way to solve QNMs is to use exact or approximated algorithms. Alternatively, one can solve QNMs by the means of simulation. In contrast to analytic algorithms, the simulation engine can estimate performance measures for models that include state-dependent real-world objectives like adaptive prioritization strategies or state-based configuration adjustments [12]. In this paper we evaluate prioritization strategies in QNMs and therefore use Discrete-Event Simulation to estimate the performance measures of the QNMS.

4.2. Simulation

In a Discrete-Event Simulation, the state of each station in the network changes only at discrete points in time, for example, when a job enters the system, a job is dispatched at a service component, or a job leaves the system after completion. Incoming jobs to a workflow are generated based on the distribution of the arrival rate on this workflow. Similarly, the event for dispatching jobs at a service component is generated using the distribution of the service time for this service component.
In order to analyze our real-world DTP systems, we make use of a custom developed discrete-event simulation engine in our experiments. The simulation engine is part of our Performance Modelling Tool suite PerMoTo [13], designed for the evaluation of DTP systems.
Figure 4 shows the structure of a single service station called *node* in the simulation. Each node comprises three parts: an input section, a server section, and an output section. The input section is responsible for receiving incoming jobs; storing them in a queueing buffer and releasing them from the queue by

realizing a certain queueing discipline, e.g. a prioritization discipline that discriminates the jobs according to their priority level. In the example of Figure 4 the third job in the input section is the job with the highest prioritization level 3 and will be served next. The service section simulates the service execution on the node. The time needed to process the job on this station is specified by the input parameters. As soon as a job finishes its execution, the outgoing section forwards the job according to pre-configured outgoing connection probabilities to a subsequent node.

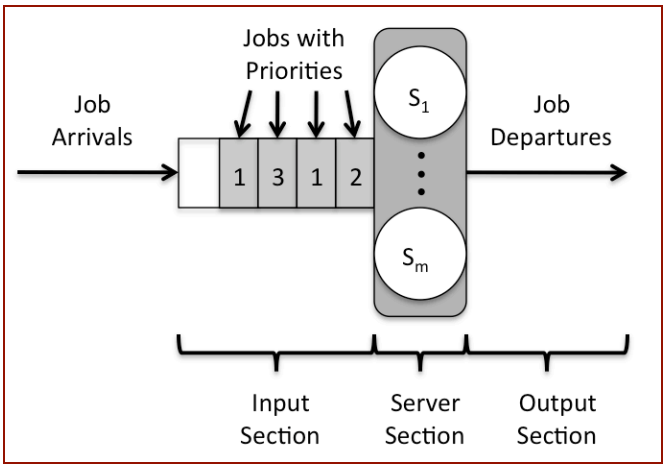

Figure 4: Simulation node implementation: input, service, and output section

The usage of prioritization in a DTP system requires that priority levels are introduced and all workflows of the system are matched into one of those priority classes. Possible hints for this classification might be the business-criticality of the workflow, the direct performance feedback of this single workflow to the customer or the risk, that a related SLA might be violated.

5. Prioritization Experiment

5.1. Characteristics of the systems under study

We analyzed the effect of prioritizing workflows that are at risk to violate SLAs due to short-term demand peaks. The effect of prioritization comes along with harms to the performance of other workflows run on the shared IT infrastructure. Therefore, we analyzed two DTP systems of a telecom provider; System Alpha and System Bravo, referred to as A and B.

System A is the central IT backbone for workflows related to the management of the retail customer segment including billing, customer data acquisition, network provisioning, and phone number management. The technical implementation is done based on the Transaction Monitor product Bea Tuxedo. Requests on the system are initiated by internet portals, shop-based applications, and call centers.

System A serves 18 business critical workflows. The individual tasks of the workflows are achieved by accessing 90 different service components. The length of the workflows varies: while a single one is very short as it consists of only three service component steps, the other ones are more complex and contain up to 53 single service calls. The length of the workflows varies as well - while one contains only three single service component steps, others call up to a maximum of 89 services. Single service component types are typically called by more than one workflow; one of the services is called by each of the 18 workflows.

System B is the integration backbone for workflows related to the management of products hosted by our industry partner but originally sold as prepaid telecommunication or DSL packages by external third-party companies. Therefore many external applications are integrated in the workflows that cannot be controlled by our industry partner. System B serves two main classes of workflows: order-entry workflows initiated by customers over a voice portal, and workflows called by internal IT systems of the partner companies like billing or tariff administration. Technically, B is based on a customized version of the TP monitor product Tibco Business Works.
Table 1 summarizes the characteristics of System A and B.

Table 1: Characteristics of Systems A and B

System characteristics	System A	System B
	Amount / Min-Max	*Amount / Min-Max*
Overall number of workflow types in system	18	15
Overall number of service components in system	90	35
Overall service components call in system	209	90
Number of service component types in single workflow	3-53	1-17
Number of service component steps in single workflow	3-89	1-19
Number of workflow types calling single service component type	1-18	1-7

5.2. Analyzed Workload Data

The overall demand but also the workflow mixes of DTP systems change dynamically. For our industry partner, for example, the weeks before Christmas are the top-selling period of the year. Thus, during this time the overall workload on the systems is significantly higher than throughout the rest of the year.
The DTP systems of our industry partner have a release cycle of three months that affects the arrival rates of all workflows handled by the system. Figure 5 shows the workload mix of the eight major workflows for three consecutive releases of one of the systems, which we analyzed in our research. The arrival rates vary independent from each other over the three releases. While the request ratios of workflow WF A1 increased, those of the other workflows declined. Hence, the trend for the arrival rate development of one single workflow needn't be correlated to the trends of the other workflows of a DTP system.

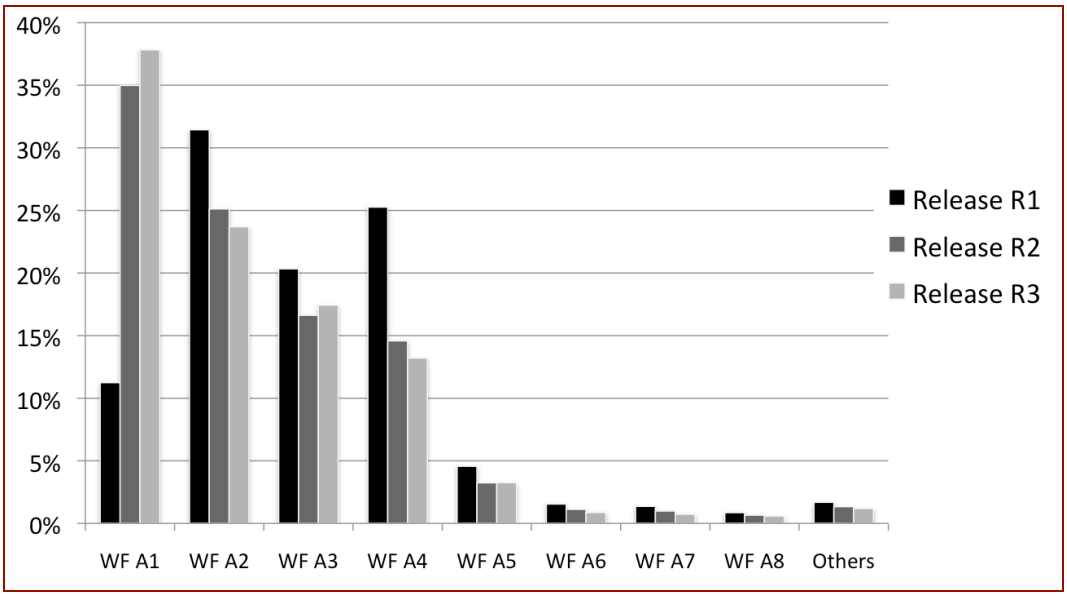

Figure 5: Workload mix shares of the workflows of DTP System A over three release cycles

While the overall amount of requests increased by 33% during Release R2, it decreased by nearly 10% during Release R3 compared to Release R1. In general such workload mix changes have different reasons, such as marketing campaigns, tariff changes, introduction of new products, activities of the competitors, etc.
Figure 6 shows the daily workload of workflow WF A2 of the same system over three weeks in June 2008. It starts on a Monday and goes until the third following Sunday. We have a daily sample workload with nightly batch jobs and a characteristic usage pattern over the day. In addition, we can observe a weekly seasonality: the workload is significantly lower on Sundays. These effects are typical for workflows on this system.

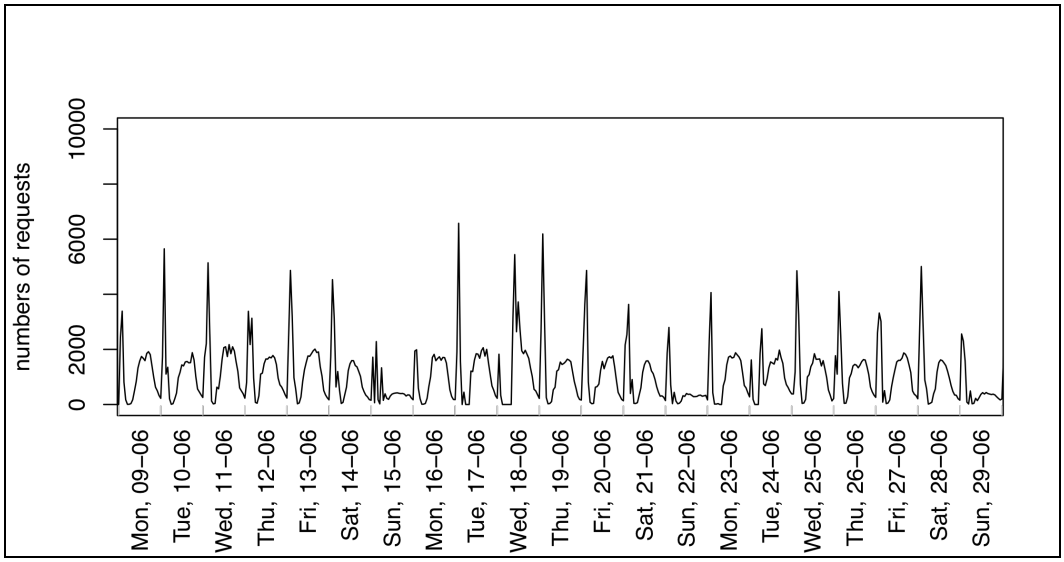

Figure 6: Three week plot of the daily sample workload of workflow on the DTP system

In order to estimate model parameters, we have analyzed the log data of 30 weekdays in June and July 2008 generated by the TP monitors. Timestamps and

request IDs in the log files allow for estimating the input parameters that are relevant for queueing network models and simulation, most notably arrival rates on a workflow level and service times on a service component level, as well as end-to-end response times. We modelled the service times and interarrival times as exponential distributions leading to M/M/c queueing networks as each service component in our systems has up to 8 parallel workers. Note that we have put considerable effort in data cleansing and outlier detection, which were due to inconsistent log syntax, rollbacks, and calls to external systems components.
We modelled the two DTP systems as QNMs with 140 and 126 service components resp. The services are provided by parallel servers: some up to eight. Apart from service components with very short response times, the maximum response time of one service component in the examined period was 5.46 seconds for system A and 2.53 seconds for system B. The mean response workflow time for system B is 5.13 seconds and much shorter than for system A (37.81 seconds). The main characteristics of the two parameterized QNMs are described in Table 2.

Table 2: Key characteristics of the Queueing Network Model of Systems A and B

Queueing Network Model Characteristics	**System A**	**System B**
	Amount / Min-Max	*Amount / Min-Max*
Number of classes (i.e., workflows)	18	15
Number of stations (i.e., services)	140	126
Number of parallel workers in station	1 - 8	1 - 6
Min - Max response time on service component level [sec]	0.01 - 5.46	0.051 - 2.53
Mean response time on workflow level [sec]	37.81	5.13
Min - Max response time on workflow level [sec]	4.41 - 248.26	0.01 - 36.06

5.3. Experimental Results

In our experiments, we evaluated three different scenarios for both DTP systems:

- A base scenario,
- an increased load scenario, and
- a prioritization scenario.

The base scenario represents the actual configuration and load of the DTP system. We therefore calculated the respective model parameters from the historic log files of the DTP systems with the help of a custom implemented Log Analyzer Tool. The increased load scenario is an extension of the base scenario where the relative workload mix remains constant but the absolute number of requests is increased for all workflows in a linear way up to 600% of the initial value. This scenario can be applied for detecting possible bottleneck candidates of the systems for higher loads. In addition to the increased load scenario, each workflow has an assigned priority level in the prioritization scenario. The respective workflow priority levels are determined with respect to the risk of violating a SLA when assuming the higher workload of the increased load scenario. Therefore, the prioritization scenario will show the impact of prioritization levels on the overall system's performance.

5.3.1. Base Scenarios

We modelled the base scenarios of the two DTP systems as open and closed QNMs in order to evaluate the predictive accuracy of both QNM types for DTP systems. In addition, we applied a DES model that matches the parameters of the open QNM. During our experiments, we focused on the prediction of the end-to-end workflow response times with the respective method. We then compared the predictions with the real-world response times calculated directly from the log files of the DTP systems. The predictive accuracy for all three models for System A was within 3% of the actual values, with a single outlier of around 11% [14]. For System B, the predictive accuracy was within 15%. Menasce et. al state in [10] that deviations of 10-20% for the predictive accuracy of response times are acceptable as they typically exhibit large variance due to system latencies.

5.3.2. Increased Load Scenario of System A

As described above, the increased load scenario models a request ratio of 600 % of the initial load of the basic scenario. This is also an indication of considerable over-provisioning of the current system. Table 3 presents the response times of the base scenario and the predicted ones of the increased load scenario for the eight most frequent workflows of System A. Furthermore, the relative deviation of the mean response times is given.

Table 3: End-to-end workflow response times of the base and the increased load scenario of the eight most frequent workflows of System A

Workflow	Base Scenario	Increased Load Scenario	Response Time Deviation when compared to the Base Scenario
	Response Time [sec]	Response Time [sec]	Deviation [%]
WF A1	226.69594	309.53597	36.54
WF A2	238.59284	442.65372	85.53
WF A3	5.37035	5.34819	0.41
WF A4	4.94283	4.94484	0.04
WF A5	178.38016	389.23448	118.21
WF A6	10.14721	10.14642	0.01
WF A7	9.32691	9.33009	0.03
WF A8	14.88074	14.87481	0.04

As one can see, three of the workflows exhibit a significantly raised response time (plus 36.54%, 85.53%, and 118.21% respectively). These workflows share some basic service components that evolve as bottleneck for the increased load scenario.

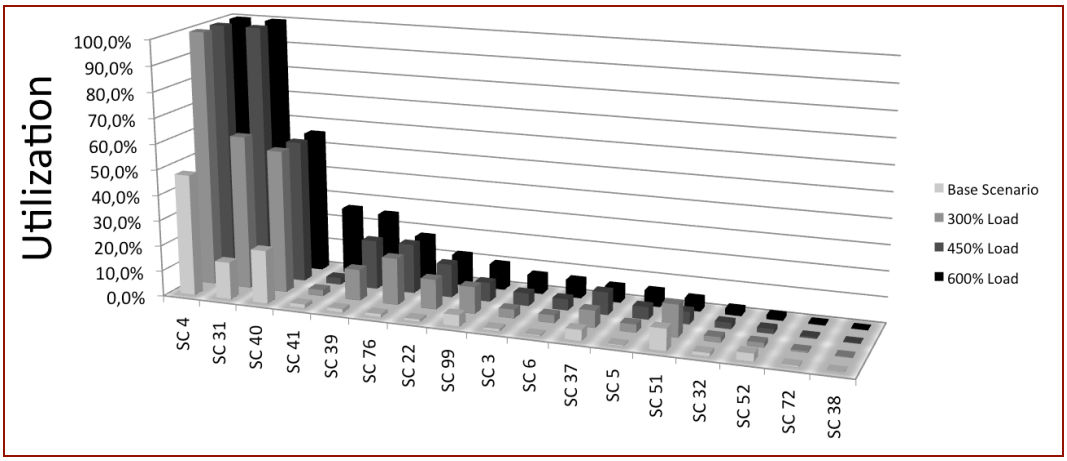

Figure 7: Utilization development of the service components (SC) included in workflow WF A2

The resulting utilization ratios of the service components called within workflow WF A2 are shown in Figure 7: when increasing the load stepwise from the base scenario up to the 600% of the increased load scenario, the utilization of service components 4, 31, and 40 increase significantly. At 300% load the utilization of Service 4 is already near 100%. Service components 4 and 31 are furthermore included in the two workflows WF A1 and WF A5 (see Figure 8). These bottleneck service components lead to the response time development of the three workflows shown above.

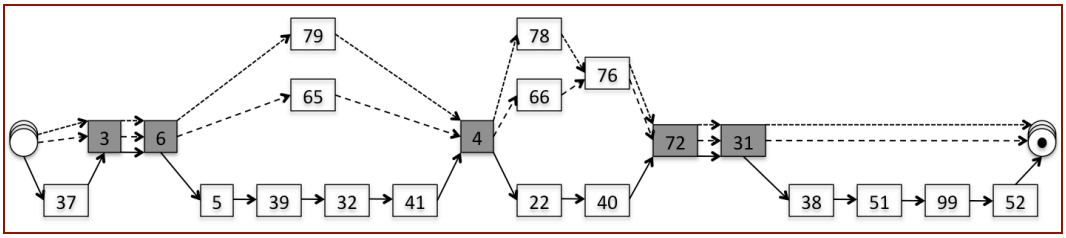

Figure 8: Workflow sequence of workflows WF A1 (dotted arrows), WF A2 (solid arrows) and WF A5 (dashed arrows)

5.3.3 Prioritization Scenario of System A

The business critical workflows of System A are the workflows WF A2, WF A6 and WF A7 as they have direct costumer interaction. The request execution finish times are recognizable directly to the customers as e.g. their phone is activated then. In order to keep the respective waiting time acceptable even in times of load peaks (or even decrease it), we apply prioritization in order to increase the performance of the affected workflow WF A2.
We use three priority levels in the model of our prioritization scenario: 1, 2, and 3, where 3 is the level with the highest priority. Consequently, workflow WF A2 gets a priority of 3 to improve its response time while the other business critical workflows get a priority of 2 to keep the response time at the same level. The other workflows of System A get the lowest priority of 1. The configured priority levels are summarized in Table 4.

Table 4: Workflow priorities in the models of the prioritization scenario for System A

System A	Business Importance	Prioritization Level
WF A1	Not critical	1
WF A2	Critical	3
WF A3	Not critical	1
WF A4	Not critical	1
WF A5	Not critical	1
WF A6	Critical	2
WF A7	Critical	2
WF A8	Not critical	1

The resulting response time behavior of workflow WF A2 for this prioritization scenario is shown in Figure 9: while the dashed line shows the respective response time of the scenario without prioritization, the solid one marks the resulting response time behavior with priority level 3. As one can see, the maximum load was increased by applying the prioritization while the response time of WF A2 was of higher performance.

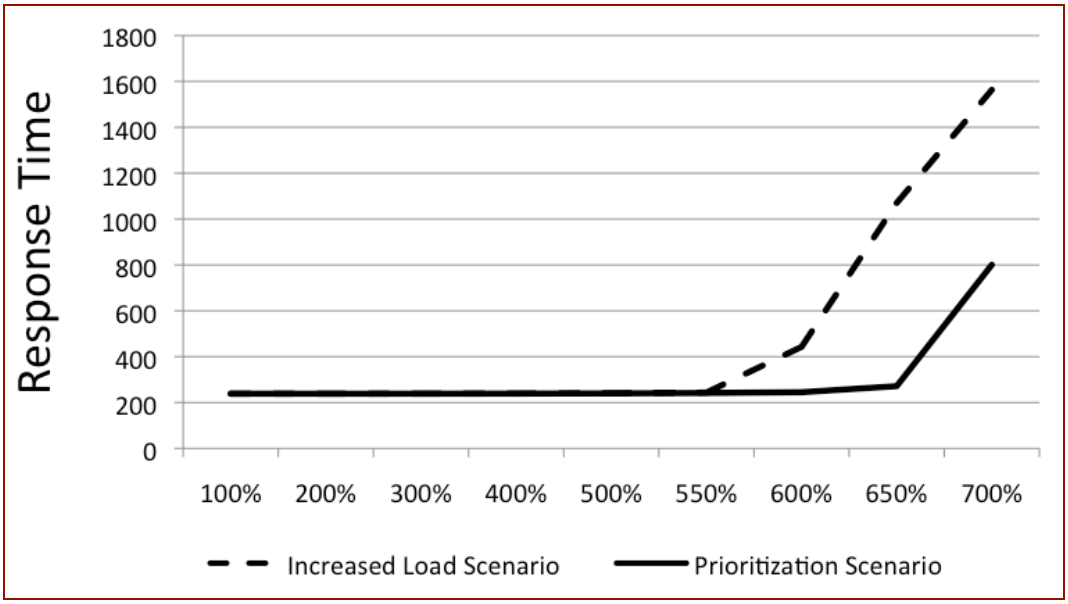

Figure 9: Response time development of workflow WF A2 when increasing the load with and without prioritization

Of course, this prioritization has side effects to the other workflows as well. For example the response times of the other two workflows containing the bottleneck stations increase. Table 5 summarizes the resulting response times and the respective deviations for the eight most frequent workflows of System A for the different scenarios. Additionally, the response time deviation for the increased load scenario and the prioritization scenario when compared to the base scenario are given.

Table 5: Workflow response times of the eight most frequent workflows of System A for the base scenario, the increased load scenario, and the prioritization scenario

Workflow	Base Scenario	Increased Load Scenario	Response Time Deviation when compared to the Base Scenario	Prioritization Scenario	Response Time Deviation when compared to the Base Scenario
	Response Time [sec]	Response Time [sec]	Deviation [%]	Response Time [sec]	Deviation [%]
WF A1	226.69594	309.53597	36.54	1746.68921	670.50
WF A2	238.59284	442.65372	85.53	244.96256	2.67
WF A3	5.37035	5.34819	0.41	5.36824	0.04
WF A4	4.94283	4.94484	0.04	4.94589	0.06
WF A5	178.38016	389.23448	118.21	3611.83304	1924.80
WF A6	10.14721	10.14642	0.01	10.14714	0.00
WF A7	9.32691	9.33009	0.03	9.32798	0.01
WF A8	14.88074	14.87481	0.04	14.88039	0.00

Figure 10 shows the deviation of the end-to-end workflow response times over the three scenarios of the three workflows that are affected most. On the left hand side, the workflow WF A2 is shown. As described above, the highest priority level 3 was assigned to this workflow. Furthermore, the workflows WF A1 and WF A5 are displayed as they suffer most from the introduced prioritization. The left bar shows the initial average response time at the base load (100%). The middle and right bars represent the predicted response times of the increased load scenario and the prioritization scenario respectively.

When compared to the base scenario, the response time of WF A2 increases by 85.5% for the increased load scenario. As effect of the prioritization scenario, the response time of this workflow is nearly reduced to the base level again despite the increased load: the response time is increased only by 2.67% when compared to the base scenario. Low prioritized workflows suffer of course from the introduced prioritization: the response time of workflow WF A1 increases from 36.5% to 670.5% when compared to the base load; the response time of WF A5 from 118.2% to even 1924.8%.

However, this increase of the end-to-end response times of the two low-prioritized workflows has no impact to the visibility of the system performance to the customers as these workflows are not business critical. The other workflows of System A are not concerned at all by the prioritization of workflow WF A2.

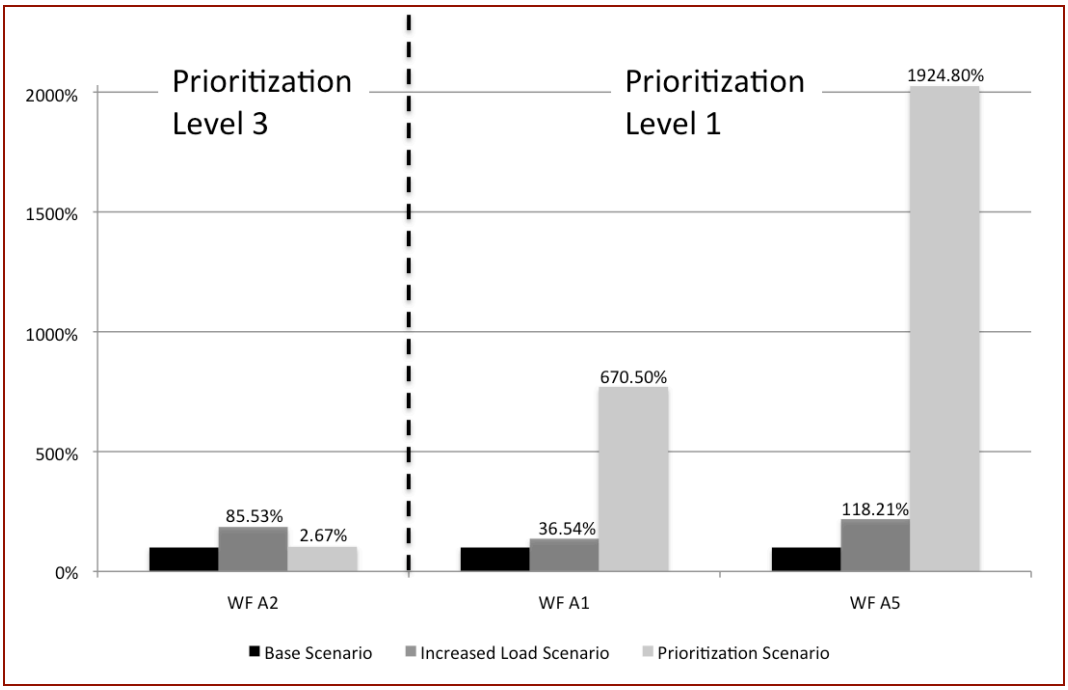

Figure 10: End-to-end workflow response time deviation of the three most affected workflows (WF A2, WF A1, and WF A5) over the three scenarios

5.3.4. Experimental Results of System B

In addition to the analysis of System A as described above, we evaluated System B in the same way. Although System B serves nearly the same number of workflows (15 compared to 18), the workflows in System B consist only of a third of different service components. That leads to more interdependencies in the workflows than in System A. Additionally many applications of external partners are included which cannot be directly affected by the prioritization in the DTP system.

For System B, we prioritized the workflow WF B4 with highest priority as it coordinates the activation of new customers. The priorities of the 8 most frequent workflows of B are shown in Table 6.

Table 6: Workflow priorities in the models of the prioritization scenario for System B

System B	Business Importance	Prioritization Level
WF B1	Critical	2
WF B2	Critical	2
WF B3	Not critical	1
WF B4	Critical	3
WF B5	Not critical	1
WF B6	Critical	2
WF B7	Not critical	1
WF B8	Not critical	1

Again, we evaluated the three scenarios as described above. Table 7 summarizes the resulting end-to-end workflow response times and the deviations to the base scenario. While six workflows seem to be unaffected by the load increase, the workflow WF B4 develops a response time deviation of 18.2%. Again, the

performance of WF B4 can be increased by the prioritization and the response time deviation decreases to 17.6%. WF B8 suffers most from the side effects of the prioritization – the response time deviation increases from 28.8% to 117.0%. The responsible IT service managers have to make the decision whether the relatively small performance gain for WF B4 justifies the performance decrease of WF B8. So, in certain situations, even prioritization might not be an option and more capacity is required to make sure that no SLAs are violated.

Table 7: Workflow response times of the eight most frequent workflows of System B for the base scenario, the increased load scenario, and the prioritization scenario

Workflow	Base Scenario	Increased Load Scenario	Response Time Deviation when compared to the Base Scenario	Prioritization Scenario	Response Time Deviation when compared to the Base Scenario
	RT [sec]	RT [sec]	[%]	RT [sec]	[%]
WF B1	0.25758	0.25674	0.327	0.25674	0.327
WF B2	0.25401	0.25317	0.328	0.25317	0.328
WF B3	0.43748	0.44141	0.897	0.44141	0.897
WF B4	13.30888	15.72755	18.173	15.65380	17.619
WF B5	0.01870	0.01869	0.075	0.01869	0.075
WF B6	0.01843	0.01842	0.075	0.01842	0.075
WF B7	0.01868	0.01868	0.000	0.01868	0.000
WF B8	8.07691	10.40152	28.781	17.52653	116.996

5.5. Summary of the Experimental Results

Prioritization can be successfully applied to react on short-term workload peaks during runtime. The overall effects of the introduced priority levels of our experiments can be seen for workflow WF A2 as depicted in Figure 11: the assignment to the highest priority level decreased the end-to-end workflow response time of WF A2 considerably. In the specific case of our experiment, the decreased workflow response time reaches nearly the time of the base scenario with a rather low workload. Furthermore, the overall load that the systems under study could handle within a certain performance level could be increased.
In System B the improvement of the prioritized workflow WF B4 is too small to justify the further increase in response time of WF B8. The cause is the complete different structure of the workflows as they integrate many external applications that cannot be controlled by the prioritization. So the use of prioritization is no general answer for short-term demand peaks of workflows in DTP systems.

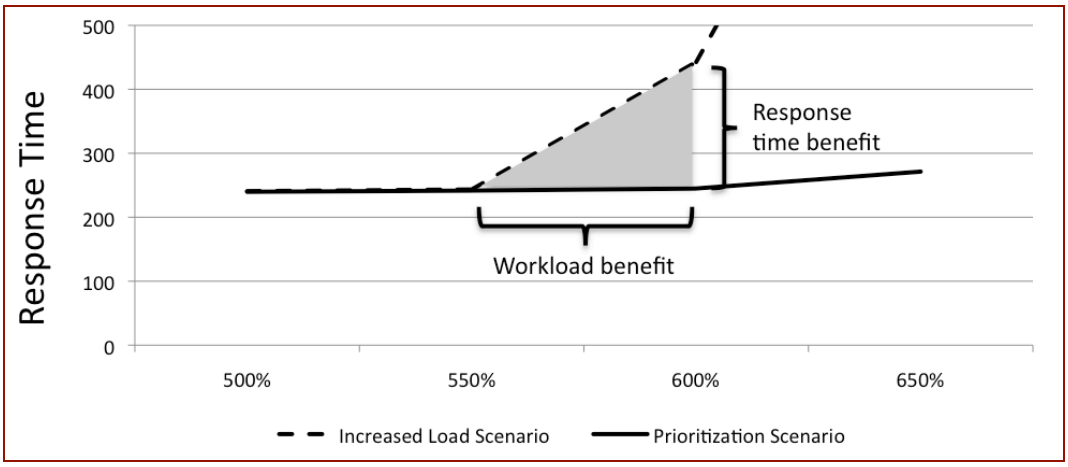

Figure 11: The introduction of the priority levels in our experiment had two dimensions for workflow WF A2: an end-to-end workflow response time benefit as well as an improvement of the overall workload that the system was able to handle

6. Related Work

For IT managers it is important to gain an understanding of the systems performance at a certain load level. To analyze the impact of different load conditions some authors have run experiments with synthetic workload generation [15]. To allow for a comparison with other systems one needs to use benchmarks, like the ones of the Transaction Processing Performance Council [16]. The characteristics of real-world demand distributions have been discussed in a number of recent publications [17-19].

For DTP systems experiments are very expensive, time-consuming or even impossible at all. The infrastructure typically involves a large number of individual components and replicating this infrastructure in a laboratory setting with realistic workloads is very expensive. Performance modelling can be seen as an alternative or complement to experiments in the lab [20, 21].

Queueing models have often been used for the performance prediction. Published applications of queueing network models (QNMs) to distributed systems that we know of are restricted to rather small applications, such as three-tier web services [10, 22-24]. Urgaonka et al. have recently applied QNMs for predicting the performance of multi-tier internet services [25]. The systems were smaller and they could apply an exact mean value analysis (MVA) algorithm to solve closed QNMs with good predictive accuracy.

We focus on the impact of workflow prioritization strategies on the performance measures of the overall DTP system. Our models are by far larger with over hundred of service components. Such models cannot be solved exactly by analytic solution methods because of the state space explosion in exact algorithms to solve queueing networks. Therefore, we solve the underlying queueing network by using a custom discrete-event simulation engine.

Starvation is a main issue for prioritization in research in other areas. Most of the attempts use two queues for two different priority levels. For example in dynamic priority queueing [26] a counter guarantees that lower jobs are served at specific points in time. Threshold based priority queueing [27] uses the queue lengths as indicator for these points of time. Other queueing strategies use more than two queues [28].

7. Conclusions

DTP systems are the IT backbone of today's services industries. Proactive capacity and performance management is important, as pre-defined quality-of-service metrics must be met. While the inherent complexity of DTP systems makes provisioning already challenging, the highly dynamic workload intensity and composition makes decisions even harder. Adaptive workload prioritization is one way to react to short-term workload peaks without the need for costly capacity over-provisioning. However, the effects of such prioritization strategies need to be known in advance in order to avoid SLA violations of the lower prioritized workflows. If prioritization is an acceptable option depends on the SLAs of other workflows.

In this paper, we show how performance modelling techniques can be used to quantify the impact of different levels of prioritization. Such analysis is important to understand the effects of different prioritization settings on the system. We analyze the workloads of two real-world DTP systems of a telecom provider. Based on the data sets we experimentally evaluated the predictive performance using discrete event simulation and the impact of several prioritization strategies. We could show that prioritization can be successfully applied to guarantee the pre-defined SLAs for the business critical workflows during peak demand times, and that queueing models provide a low-cost method to analyze the impact of prioritization.

In our future work we want to develop strategies and models for the automated detection of workload peaks during runtime. Proactively evaluated prioritization strategies can then be applied in order to overcome such short-term workload peak.

9. Acknowledgements

The project was funded by means of the German Federal Ministry of Economy and Technology under the promotional reference "01MQ07012". The authors take the responsibility for the contents.

10. References

[1] Oracle Corporation (2009) Oracle Tuxedo, Oracle Data Sheet, http://www.oracle.com/products/middleware/docs/tuxedo-datasheet.pdf, Online accessed 2010-04-19

[2] Vitria Technology (2006) Business Ware, Business Process Integration for SOA & Event Driven Architectures, http://www.vitria.com/wp-content/download/BW_Brochure.pdf, Online accessed 2010-04-19

[3] TIBCO Software (2008) TIBCO ActiveMartix BusinessWorks, http://www.tibco.com/multimedia/ds-businessworks_tcm8-805.pdf, Online accessed 2010-04-19

[4] Markl C, Hühn O (2009) Evaluation of Prioritization in Performance Models of DTP Systems. In Proceedings of the 11th IEEE Conference on Commerce and Enterprise Computing (CEC), Vienna, Austria

[5] Ruh WA, Brown WJ, Maginnis FX (2000) Enterprise Application Integration: A Wiley Tech Brief. John Wiley & Sons, Inc, New York, USA

[6] Distributed Transaction (1991) The XA Specification. X/Open Company Ltd.

[7] Wilson JH, Keating B (2002) Business Forecasting with Accompanying Excel-Based ForecastX™ Software, McGraw-Hill, New York, USA

[8] Winters PR (1960) Forecasting Sales by Exponentially Weighted Moving Averages. In INFORMS

[9] Bailey DH, Snavely A (2005) Performance Modeling: Understanding the Present and Predicting the Future. In: Euro-Par 2005 Parallel Processing, Lisbon, Portugal

[10] Menasce DA, Dowdy LW, Almeida VAF (2004) Performance by Design: Computer Capacity Planning By Example. Prentice Hall PTR, Upper Saddle River, NJ, USA

[11] Bolch G et al (2006) Queueing Networks and Markov Chains - Modeling and Performance Evaluation with Computer Science Applications. Second ed. Hoboken, John Wiley & Sons, Inc., New Jersey, USA
[12] Fishman GS (2001) Discrete-event simulation: modeling, programming, and analysis. Springer Verlag
[13] Hühn O, Markl C (2007) PerMoTo - Performance Modelling Tool suite. In WITS 07 - Seventeenth Annual Workshop on Information Technologies and Systems, Montreal, Canada
[14] Hühn O, Markl C, Bichler M (2009) On the predictive performance of queueing network models for large-scale distributed transaction processing systems. In Information Technology and Management, Vol. 10, No 2-3, pp 135-149
[15] Krishnamurthy D (2006) A synthetic workload generation technique for stress testing session-based systems. In IEEE Transactions on Software Engineering, Vol. 32(11), pp 868-882
[16] TPC (2009) TPC Transaction Processing Performance Council, http://tpc.org, Online accessed 2009-10-17
[17] Feitelson DG (2002) Workload Modeling for Performance Evaluation. In Performance Evaluation of Complex Systems: Techniques and Tools, Vol. 2459/2002, pp. 114-141
[18] Feitelson DG (2002) The Forgotten Factor: Facts on Performance Evaluation and its Dependence on Workloads. In Euro-Par: Proceedings of the 8th International Euro-Par Conference on Parallel Processing, London, UK
[19] Stewart C, Kelly T, Zhang A (2007) Exploiting Nonstationarity for Performance Prediction. In EuroSys: Proceedings of the 2nd ACM SIGOPS/EuroSys European Conference on Computer Systems, New York, USA
[20] Noel E, Tang KW (2000) Performance modeling of multihop network subject to uniform and nonuniform geometric traffic. In IEEE/ACM Transactions on Networking (TON)
[21] Thakkar SS, Schweiger M (1990) Performance of an OLTP Application on Symmetry Multiprocessor System. In 17th Annual International Symposium on Computer Architecture, Seattle, WA
[22] Mazzucco M, Mitrani I, Palmer J, Fisher M, McKee P (2007) Web Service Hosting and Revenue Maximization. In Proceedings of the Fifth IEEE European Conference on Web Services (ECOWS), Halle, Germany, pp. 45-54
[23] Chen Y et al (2007) SLA Decomposition: Translation Service Level Objectives to System Level Thresholds. In ICAC ´07: Proceedings of the Fourth International Conference on Autonomic Computing : IEEE Computer Society, Washington, DC, USA
[24] Steward C, Shen K (2005) Performance Modeling and System Management for Multi-component Online Services. In Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation, Vol. 2, pp. 71-84
[25] Urgaonkar B et al (2005) An analytical model for multi-tier internet services and its applications. In SIGMETRICS ´05: Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems, New York, USA
[26] Ghanwani A, Gelenbe E (1998) Approximate Analysis of a Dynamic Priority Queueing Method for ATM Networks. In PICS 98 - Seventh International Conference on Performance of Information and Communication Systems, Lund, Sweden
[27] Lee D, Sengupta B (1993) Queueing Analysis of a Threshold Based Priority Scheme For ATM Networks. In: IEEE/ACM Transactions on Networking, Vol. 1, No 6., pp 709-717
[28] Katayama T, Kobayashi K (2007) Analysis of a nonpreemptive priority queue with exponential timer and server vacations. In Performance Evaluation, Vol. 64, pp 495-506