Chapter 19
Model-Driven Performance Evaluation of Web Application Portals

Nilabja Roy
Vanderbilt University, USA

Douglas C. Schmidt
Vanderbilt University, USA

ABSTRACT

Web application portals cater to various types of concurrent users and requests. The number of requests varies by time of day and day of week. Despite the variation in workload, it is important to provide the expected performance (response time) to users of these applications. To assure an appropriate level of performance, web application portals should be analyzed and evaluated throughout their software development lifecycles. Model Driven Architecture (MDA) provides a structured process for developing and analyzing web application portals from the requirement analysis to the ultimate deployment. This chapter examines recent advances in performance analysis methods for web application portals and shows how they can be integrated with MDA methods to analyze performance analysis throughout their software development lifecycles.

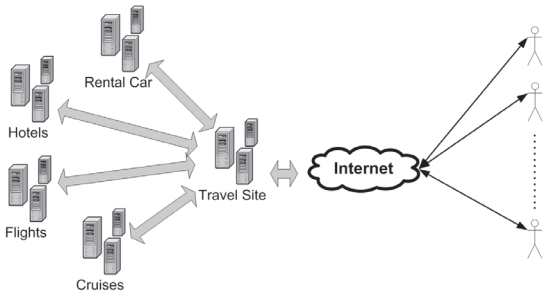
INTRODUCTION

Emerging trends and challenges. The advent of web-based applications, such as shopping, social networking, photos, videos, music, gaming, and chat, are increasing the popularity and accessibility of the Internet. There is also growing focus on application integration platforms, such as Sun's Java Composite Application Platform Suite, Facebook's Application Platform, and Oracle's Application Development Framework, where a single portal can provide many services. These integrated web sites are referred in this paper as *web application portals*, which are Internet sites that provide multiple services to users. For example, users of social networking sites, such as Facebook (www.facebook.com) and MySpace (www.myspace.com), upload recent photos and videos, exchange messages and chat with each other, and play online games with friends.

Figure 1 shows the architecture of a typical web application portal, such as www.priceline.

DOI: 10.4018/978-1-61692-874-2.ch019

Figure 1. A travel site that provides interface to hotels, flights, rental cars, and cruises



com or www.hotwire.com, that help users build vacation packages with choices for flights, hotels, rental cars, and cruises. Users submit requests to the portal, which in turn contacts various service providers for each service and forwards responses to users. Web application portals should be scalable to support a variety of services and the large number of customers accessing the services simultaneously. The scalability requirements for these sites typically grow as the number of service providers increases.

Competition between providers of web application portals is also growing. Providers have different marketing differentiators and focus on different features and services. For example, IMDb (www.imdb.com) focuses on detailed movie information, whereas yahoo-movies (www.movies.yahoo.com) focus on movie ratings and theater showings. Although both features are useful, they serve different sets of users with somewhat different interests. With multiple services, differentiated focus, and large customer bases, such web portals are a complex composition of different sources of non-determinism, which complicates the evaluation of web application portal performance.

Regardless of the features and services offered by web application portals, successful providers must ensure key quality of service (QoS) properties, such as end-to-end request response time, system availability, and scalability. For example, online travel sites, such as Expedia(www.expedia.com) or Orbitz(www.orbitz.com), aim to provide the best travel deals to customers within a reasonable time frame, which may vary from person-to-person and from application-to-application. In particular, user submitting travel queries may be willing to wait longer for the best deals than users searching for a phone number. Given the proliferation of web-based application portals in the Internet, users who are not satisfied with one provider can often switch to alternative providers, which incentivizes providers to enhance the QoS of their web application portals.

To remain viable in today's competitive environment, therefore, developers and administrators of web application portals must address a number of issues, such as (1) what software/hardware architecture will provide the necessary performance at scale, (2) how should the software be modularized, (3) how can applications and systems be configured to ensure high performance, (4) how particular application design performs under certain usage patterns, and (5) how many (and what type) of machines are required to achieve the required performance. Addressing these issues can help developers of web application portals design systems that can provide the required QoS for current and planned usage models.

In many cases, however, web portal application performance is evaluated late in the system lifecycle, i.e., after the software is developed and deployed on the target hardware. At this point, it is hard to correct mistakes in the system design that yield poor performance. What is needed, therefore, are techniques that analyze and predict the performance of web-based applications earlier in the lifecycle and can help guide developers choices of alternative portal designs. These techniques need not provide exact predictions, but they do need to accurately capture general trends and provide quantifiable numbers that enable developers to select the most appropriate alternative designs.

Developers and administrators must address various challenges to ensure that web portal applications meet their QoS requirements. For example,