



Chapter IV

Approaches to Building High Performance Web Applications: A Practical Look at Availability, Reliability, and Performance

Brian Goodman, IBM Corporation, USA

Maheshwar Inampudi, IBM Corporation, USA

James Doran, IBM Corporation, USA

Abstract

In this chapter, we introduce five practices to help build scalable, resilient Web applications. In 2004, IBM launched its expertise location system, bringing together two legacy systems and transforming the employee's ability to find and connect with their extensive network. This chapter reviews five of the many issues that challenge enterprise Web applications: resource contention, managing transactions, application resiliency, geographic diversity, and exception perception management. Using the IBM expertise location system as context, we will present five key methods that mitigate these risks, achieving high availability and high performance goals.

Introduction

In this chapter, we introduce five practices for building scalable, resilient Web applications. First, we briefly review the context in which IBM launched its internal expertise location system in 2004. We then introduce the challenges we faced in implementing the business requirements and present five key methods that mitigated risks to achieving our high availability and performance goals.

Specifically, we will look at:

- caching strategies for high availability Web applications: beyond storing copies of HTML (Challenger, Dantzig, & Iyengar, 1998; Iyengar & Challenger, 1997);
- asynchronous task processing within Web applications: removing non-essential linear logic from high-volume transactions (Grand, 2002);
- building self-reliant autonomous behavior: encapsulating through services achieving tight integration with *true* loose coupling (Birman, van Renesse, & Vogels, 2004);
- client-side Model View Control (MVC): moving MVC to the browser supercharging the response times and getting a “wow” user experience (Murry, 2005; Sun Microsystems, 2002);
- graceful degradation: keeping users thinking and feeling “fast,” “reliable,” and “always on” (Florins & Vanderdonckt, 2004).

Caching Strategies

Caching strategies are a core part of high-performing Web experiences. In many cases (Amiri, Park, & Tewari, 2002; Candan, Li, Luo, Hsiung, & Agrawal, 2001; Liebmann & Dustdar, 2004; Rodriguez, Spanner, & Biersack, 2001), the assumption is that caching occurs at the edge of the network, the closest point to the consumer and the furthest from the data or application. Another somewhat overlooked approach is object caching.

Davison (2001) provides a wonderful primer on Web caching that illustrates the principles of caching and highlights some of the issues that may arise due to its use. Edge caching, or Web caching, is focused on storing and managing Web pages (static or dynamic) to help speed up transaction times. As the complexity of Web architecture evolves, applications have become more distributed. Edge solutions exemplify this, placing caches of content, sometimes fragments of executable code, in multiple geographical locations.

In recent years, object caching has enjoyed a revival and is now seen as a more desirable component of Web application architecture. Caching objects and sharing them across an infrastructure is a compelling capability. Often, the cost associated with building an object is considered quite high. The difference with an object cache mechanism is that it often resides at the Web server (Jadav & Gupta, 1997), the Web application, or as a middleware between the data and the Web application logic. Once an object is built, it can be cached, distributed, and managed for future reuse; it delivers performance at the application layer, whereas edge caching offers performance benefits to the delivery of data.

33 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/approaches-building-high-performance-web/5217

Related Content

A Methodology for Integrating Patterns in Quality-Centric Web Applications

Pankaj Kamthan (2008). *International Journal of Information Technology and Web Engineering* (pp. 27-44).

www.irma-international.org/article/methodology-integrating-patterns-quality-centric/2645

Relationship Analysis: A Technique to Enhance Systems Analysis for Web Development

Joseph Catanio and Michael Bieber (2005). *Web Engineering: Principles and Techniques* (pp. 97-113).

www.irma-international.org/chapter/relationship-analysis-technique-enhance-systems/31109

Dimensions of Business-to-Consumer (B2C) Systems Success in Kuwait: Testing a Modified DeLone and McLean IS Success Model in an E-Commerce Context

Kamel Rouibah, Paul Benjamin Lowry and Laila Almutairi (2016). *Web-Based Services: Concepts, Methodologies, Tools, and Applications* (pp. 1223-1255).

www.irma-international.org/chapter/dimensions-of-business-to-consumer-b2c-systems-success-in-kuwait/140849

A Subspace Clustering Framework for Research Group Collaboration

Nitin Agarwal, Ehtesham Haque, Huan Liu and Lance Parsons (2006). *International Journal of Information Technology and Web Engineering* (pp. 35-58).

www.irma-international.org/article/subspace-clustering-framework-research-group/2602

E-mail Mining: Emerging Techniques for E-Mail Management

Ioannis Katakis, Grigorios Tsoumakas and Ioannis Vlahavas (2007). *Web Data Management Practices: Emerging Techniques and Technologies* (pp. 220-243).

www.irma-international.org/chapter/mail-mining-emerging-techniques-mail/31103