Username: Pralay Patoria **Book:** Pro .NET Performance. No part of any chapter or book may be reproduced or transmitted in any form by any means without the prior written permission for reprints and excerpts from the publisher of the book or chapter. Redistribution or other use that violates the fair use privilege under U.S. copyright laws (see 17 USC107) or that otherwise violates these Terms of Service is strictly prohibited. Violators will be prosecuted to the full extent of U.S. Federal and Massachusetts laws.

CHAPTER 11

Web Application Performance

Web applications are designed to handle hundreds and even thousands of requests per second. To build such applications successfully, it is important to identify potential performance bottlenecks and to do everything you can to prevent them from occurring. But handling and preventing bottlenecks in ASP.NET applications is not confined to your code alone. From the time a web request reaches the server to the time it reaches your application's code, it passes through an HTTP pipeline and then through the IIS pipeline, only to reach another pipeline, the ASP.NET's pipeline, and only then does it reach your code. And when you finish handling the request, the response is sent down those pipelines until it is eventually received by the client's machine. Every one of these pipelines is a potential bottleneck, so improving the performance of ASP.NET applications actually means improving both *your* code and the performance of the pipeline.

When discussing ways to improve the performance of ASP.NET applications, one has to look further than just the application itself, and examine the way various parts that make a web application affect its overall performance. The overall performance of a web application is comprised of the following:

- The application's code
- The ASP.NET environment
- The hosting environment (in most cases, IIS)
- The network
- The client-side (not discussed in this book)

In this chapter we will briefly discuss tools for performance testing of web applications and explore various ways, from each of the above mentioned subjects, which can help us improve the overall performance of our web application. Toward the end of the chapter, we will discuss the need and implications of scaling web applications and how to avoid known pitfalls when scaling.