

## Foreword

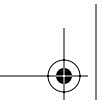
Eclipse is a game-changing industry phenomenon. Launched in open source in the fall of 2001, Eclipse has come a long way from its roots as an internal project at IBM's OTI subsidiary—designed originally as a way for IBM to integrate its desktop tools onto a common software base. In 2004 the first ever EclipseCON conference in Anaheim, California, was an overwhelming success, and the Eclipse Foundation was launched that year as the independent entity overseeing the future evolution of the technology in open source. Eclipse truly has taken flight on its own wings.

Why has Eclipse had such an impact on the tools industry? The answer is because it's great technology, it's open source, it has phenomenal support from a worldwide community of developers, and a whole lot of important industry players are using it to build great products.

The Eclipse vision is different—it's a “platform-centric” rather than “tool-centric” way of thinking. The bare-bones Eclipse Platform is essentially a “Universal IDE”—an IDE for anything and nothing in particular, all at the same time. When you want to build a new tool for Eclipse, you think about how you “teach” the Eclipse Platform about your problem, rather than about how you can bolt a monolithic “tool” on top of it. And the way you teach Eclipse about your problem is by writing tool plug-ins that hook into well-defined plug-in points. The result is that rather than see a new “tool” added to Eclipse, users instead see new capabilities that the Eclipse Platform is now able to perform. Different people, different teams, different organizations can build different tools at different times, yet when they are assembled into the Eclipse Platform on the user's desktop they behave—if it's done right—as if they were designed to provide a single integrated experience.

The best tools integrate so seamlessly you can't tell where one tool starts and another ends. Of course, it's possible to get this wrong and build tools





that *don't* fit well with Eclipse and don't work well with other Eclipse-based tools! To get this right you need to understand how your tool should extend Eclipse to solve your particular problem—how your tool should “plug in” to the integration points defined by Eclipse. You also need to decide when to invent something new and when you should just extend or enhance existing function. That's why this book is such an indispensable reference, not only for tool builders and rich client application developers but also for anyone customizing or extending the Eclipse environment.

Last time I checked, the first edition of this book had an *average* rating of five stars at Amazon.com—not at all surprising given the book's heritage. Fully updated and revised for Eclipse 3.0, this book was written by members of the IBM Eclipse Jumpstart team from their experience teaching courses, and it is the definitive Eclipse reference. Not only do they set out to show you how to *use* Eclipse proficiently, they then transition nicely into a comprehensive, authoritative guide to extending the environment yourself and building your own Eclipse-based tools. The examples and exercises included have been proven in real-life situations and will get you up and running quickly.

The idea of Eclipse has captured the imagination of developers from all corners of the globe—it has caught on beyond anyone's wildest dreams. But the real value of Eclipse comes not from Eclipse itself, but from what people do with it and how they extend it. It's limited only by your imagination!

—Dave Thomson  
Eclipse Project Program Director, IBM