

Foreword

by Jonathan Schwartz

Great technology stays just that—great technology—without great tools. Great tools enable great technology to reach global audiences, transform industries, and, quite literally, change the face of the Internet. And now you understand our view on the importance of NetBeans—to Sun, to the Internet, and to the user community at large.

Now, every product or network service starts with an idea. Turning ideas into reality is what NetBeans has been all about, ever since a very small team in Prague joined a bigger team in Menlo Park, California—and an even bigger team across the globe. From that small beginning, among a small but loyal community of users and developers, NetBeans has become the fastest growing multiplatform development environment we've ever seen.

The growth has been tremendous—a result of fierce competition, innovation, reinvention, and commitment; and we see no limit to the growth across the world.

As I said back in the days when I was running Sun's Developer Tools group, there are two ways to understand the strategy of a technology company. The first is to see how they compensate their sales force, to understand their immediate tactical priorities. The second is to look at the roadmap for their developer tools, to understand their longer-term, strategic priorities.

To that end, there is no product at Sun that better represents the future we envision than NetBeans. And on behalf of Sun, as just one member among many in the community now chartered with its evolution, and as one among many in the corporate user communities, I can say without reservation that it's a thrilling future indeed.

Jonathan Schwartz

Chief Executive Officer and President of Sun Microsystems, Inc.

blogs.sun.com/jonathan

Foreword

by Jan Chalupa

I started using NetBeans Developer 2.0 in the late 1990s, but I didn't care about its internals until I joined Sun and the NetBeans team in 2000 to work on NetBeans 3.0. I was coming from the world of Win 32 APIs, MFC, and COM, was moderately familiar with Java libraries and Swing, read the "Gang of Four" bible and all kinds of other books on design patterns and object-oriented programming. In many aspects, what I found in the NetBeans APIs didn't resemble anything I was used to. "What kind of pattern is this `Cookie` thing?" I wondered. "Why is the class that represents a simple view or window called `TopComponent`?" "What is the difference among `FileObjects`, `DataObjects`, and `Nodes`?" "SplittedPanel? Doesn't sound like correct grammar to me." And surprisingly, despite the prevalent code hacking and antiauthority culture inherent to the NetBeans core team, almost everything in NetBeans was accessible through a singleton class called `TopManager`.

However, soon I started to find out that no matter how weird some of the names or concepts could seem, NetBeans was architected with reusability and extensibility in mind and allowed developers to add new features easily—or even build their own applications by reusing NetBeans core classes as a framework. I came to realize that NetBeans wasn't just an IDE, but also a very powerful concept that could save application developers years of development time. NetBeans was a platform.

I also figured out why some of the building blocks looked unfamiliar and a little awkward at first glance. NetBeans started as a students' project in the mid-nineties. Most of the developers, including the architects, were university students or fresh graduates with very little experience in software design. They worked extremely hard while learning on the fly. Sometimes inventing the wheel, sometimes introducing new names for existing things, and sometimes, admittedly, making design mistakes. In spite of all this, the original idea of an extensible application platform implemented in Java turned out to be very smart, innovative, and forward-looking.

NetBeans™: The Definitive Guide (O'Riley), written in 2001–2002 and the only comprehensive book on NetBeans APIs to date, was the first attempt to make the NetBeans Platform available to a wider developer audience. Unfortunately, it was the time when the most serious architectural flaws began to emerge and became blockers for future development of the platform. By the time *NetBeans™: The Definitive Guide* was published, some of the APIs described in the book were gone and new APIs were introduced. The primary focus had shifted to making a really solid IDE, while the platform evangelization had been put on the back burner, known only to those who were really close to the NetBeans developer community.

Nevertheless, the NetBeans Platform did not disappear. Over several years, it just got better and more mature. `SplitPanel` got deprecated. So did `TopManager`, replaced with the `Lookup` concept allowing for feature discoverability and intermodule communication in a distributed and loosely coupled modular architecture. Many APIs got polished and stabilized. NetBeans IDE 5.0 added extensive support for developing modules and building applications based on the NetBeans Platform. Creating a new module became simpler than ever before. The `platform.netbeans.org` site was established and became a valuable source of documentation, articles, and tutorials about the platform.

The only thing that was still missing was a new book. I would like to thank Tim, Jarda, Geertjan, and many other contributors for filling this gap. I believe it will make the NetBeans Platform accessible to many new developers.

Jan Chalupa
NetBeans Director