

Boost efficiency and ROI in J2EE application development



OptimalJ™ Architecture Edition

Delivering reliable applications faster

Compuware OptimalJ is an advanced development environment that reduces the time, cost, risk and complexity of delivering and maintaining J2EE applications. Through model-driven, pattern-based development, OptimalJ cuts the time to produce a working application by 40 percent compared to code-centric methods.

OptimalJ bridges the gap between business and technology, so business requirements, not technical constraints, drive your development process. It simplifies maintenance, increases flexibility, promotes development best practices and leverages your existing technology for greater ROI.

OptimalJ is the only enterprise development environment which fully implements Model Driven Architecture, as standardized by the Object Management Group (OMG). It provides technology patterns which automate the transformation of business models into application models, and implementation patterns which automate the transformation of application models into code. Collectively, technology patterns and implementation patterns are called transformation patterns.

OptimalJ Architecture Edition offers architects the ability to add to or extend the patterns used by OptimalJ. It allows software architects to create or customize transformation patterns, providing a code generation framework to facilitate deployment, a model for describing the implementation patterns and a pattern language to simplify coding.

Software architects usually have clear ideas on how to implement a given model, based on internal standards, guidelines and best practices, industry standards, past experience and existing libraries of code.

The use of patterns lets you make those ideas the basis of J2EE application development throughout your IT organization, delivering maximum business value.

With OptimalJ Architecture Edition you can:

- transfer your knowledge into your application architecture automatically
- make best use of experienced developers
- build applications faster—and better
- ensure consistency across multiple developers and applications
- generate working applications using the best design and coding practices.

OptimalJ Architecture Edition gives skilled Java architects three ways to customize transformation patterns for even more effective development: building your own patterns with the Pattern Editor, extending existing patterns through the use of Join Points and editing the patterns at the source-code level.

Pattern Editor

The OptimalJ Architecture Edition introduces a new, leading-edge technology called the Pattern Editor, which allows architects to change, extend and influence OptimalJ's default technology and implementation patterns according to their company's own best practices and standards.

Compuware also offers OptimalJ in two other editions:

OptimalJ Professional Edition

provides development teams the full functionality of model-driven and pattern-based development, as well as allowing designers to customize J2EE applications to better meet business requirements.

OptimalJ Developer Edition

provides Java developers with the tools to easily customize and extend n-tier J2EE applications generated with OptimalJ without overwriting the generated code. It boosts developer productivity, relieves the burden of tedious and repetitive coding and ensures adherence to standards and guidelines.

OptimalJ's repository API allows architects to access application and domain model definitions. The repository API is derived directly from the OptimalJ metamodels. Architects may use OptimalJ's Template Pattern Language (TPL) or Java language for implementing or customizing implementation patterns.

The TPL enables architects to create the skeleton code of a generated application using the same syntax and structure as the target code itself. TPL communicates with the repository API used by OptimalJ, and this allows architects to use sophisticated features of the API declaratively.

OptimalJ's pluggable pattern architecture helps to dramatically reduce the cost of upgrading to new technologies. OptimalJ detects versions of EJBs and other Java entities at startup time and implements new features automatically, preserving all custom code.

Join Points

To ease the development of patterns, the OptimalJ Architecture Edition contains a solution called Join Points, which increases flexibility in J2EE development. The Join Points allow architects to create their own patterns and hook them into the OptimalJ standard patterns. In doing so architects can not

only extend OptimalJ's patterns, but also override undesired functionality. In addition, when Compuware delivers newer versions of these patterns, the Join Points (interfaces) will be stable and thus will work with the custom-built patterns.

Join Points enable architects to influence the code generated by OptimalJ at a number of levels. For example, they can use Join Points to replace files generated by OptimalJ with files of their own design, or to insert code from their own pattern into files created by OptimalJ.

Compuware products and professional services—delivering quality applications
Compuware is a leading global provider of software products and professional services which IT organizations use to develop, integrate, test and manage the performance of the applications that drive their businesses. Our software products help optimize every step in the application life cycle—from defining requirements to supporting production service levels—for web, distributed and mainframe platforms. Our services professionals work at customer sites around the world, sharing their real-world perspective and experience to deliver an integrated, reliable solution.

OptimalJ basics

Model-driven development

Modeling instead of programming allows architects to set standards for development teams.

Business Rules Editor

Using the Business Rules Editor, development teams can add both static and dynamic business rules at the model level.

Pattern-driven generator

OptimalJ uses patterns to generate all the application code required for a running application.

Active synchronization

OptimalJ synchronizes the Java code with the application models so that at all times the model accurately represents the actual application that is delivered.

Integrated deployment

OptimalJ automatically deploys to many of the leading J2EE production servers, and supports and automates integration with existing Web services and the creation of new ones.

The key aspects of OptimalJ technology are discussed in the OptimalJ community forums. To learn more, or to share your experiences with OptimalJ, please join JavaCentral, the Java developers community, at <http://javacentral.compuware.com>.

Please contact us to learn more about how our comprehensive products and services can help your organization improve productivity, create higher quality applications and ensure performance in production.

All Compuware products and services listed within are trademarks or registered trademarks of Compuware Corporation. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems Inc. in the United States and other countries. All other company or product names are trademarks of their respective owners.
© 2003 Compuware Corporation

