When E-SoftSys learned of OptimalJ, a model driven architecture-based tool that accelerates Java development by generating working applications directly from visual models, the company was intrigued, to say the least. OptimalJ provides a simple, yet powerful way to quickly build or modify distributed Java applications. Using patterns and model-driven application design, OptimalJ delivers high productivity and consistency throughout the development cycle, while shielding development teams from the complexities of the J2EE architecture.

**Putting OptimalJ to the Test**

E-SoftSys' E-Office internal workflow application project features an E-Maintenance module, which logs and tracks in-house systems maintenance problems. "E-Maintenance helps our employees log maintenance requests, prioritize and track the progress," President and CEO Kat Shenoy explains. "The module also produces statistics for machine downtimes, which management can use to take the appropriate corrective actions for improvement. It also helps us meet the requirements mandated by ISO certification." ISO certification requires E-SoftSys to demonstrate continuous improvement in reducing the downtimes of PCs, servers and its network.

Management was pleasantly surprised by the results: One business analyst was able to redevelop the entire E-Maintenance module in a little over half the time originally slated for the project using two experienced developers to create code manually. E-SoftSys improved productivity by 40 percent with OptimalJ.

A Development Tool That Gets the Job Done

Adopting OptimalJ and its model-driven, pattern-based development paradigm resulted in many advantages for E-SoftSys. In addition to increased developer productivity, the company also reaped the benefits of a reduced programming skills requirement, simplification of the development process, easy application maintenance and enhancement, and reduced testing requirements.
“OptimalJ helped us reduce system downtimes, making our company more efficient.”  
Kat Shenoy, President and CEO, E-SoftSys

Firm Believers in the Power of OptimalJ

When put to the test, OptimalJ delivered much more than E-SoftSys thought was possible. “In my opinion, OptimalJ is the only tool currently available that realizes the true benefits of Object Management Group’s Model Driven Architecture,” Kat Shenoy says in reference to the standard embraced by OptimalJ. “Initial manual development of the E-Maintenance application took 400 hours using two experienced Java programmers. And this did not include the additional time required to manually test the module!

“With OptimalJ, however, one business analyst, with less Java experience and purely a generalist in workflow-based applications, completed redevelopment of the same module, including testing, in 248 hours. The productivity gains and cost savings realized with OptimalJ are substantial.”

A nother benefit company representatives agree on is the quality of the working code OptimalJ produces. Rather than use two testers, as allocated in the original project, E-SoftSys found it necessary to assign only one to test the new application. “The main reason for this,” Kat Shenoy says, “is that OptimalJ produces bug-free code and testing was mostly related to functionality issues rather than finding bugs.” Nanda Kumar, Java development center manager, agrees: “OptimalJ generates excellent code.”

Taking Applications to the Next Level

Future plans for using OptimalJ at E-SoftSys include redeveloping the remaining modules of the E-Office application. According to Kumar, top on the list is the internal audit module.

“Improving efficiency and productivity is important for any company. But for E-SoftSys, constant striving to be the best is part of maintaining its ISO certification. With OptimalJ, the company can take full advantage of all its development resources, and free up its more experienced Java architecture experts to set its development standards.”

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