



"Trend is a typical example of how development teams can use FireStorm/DAO to jumpstart Java development in an organization. Data persistence architecture is usually one of the first major items that has to be dealt with. With FireStorm/DAO, Trend can ensure that best practices are consistently used from the start."

Andy Grove, CTO,  
CodeFutures Software

## **Trend Updates Financial Applications using CodeFutures FireStorm/DAO for its Java Persistence Architecture**

Trend S.p.a., the Italian leader in banking and financial software applications, has chosen FireStorm/DAO to provide its Java persistence architecture.

Trend is the Italian leader in developing and delivering financial software applications. Trend offers a portfolio of software solutions covering all operational and strategic issues specifically tailored to banks and insurance companies. The range of applications includes customer management, business control and planning, and regulatory reporting. Trend's customers include 19 of the top 21 major banks in Italy. The customer base was built up over a long period, with COBOL being the primary programming language. While Trend's customers were satisfied with their solutions, Trend recognized the need to move to a more modern programming language and development environment. Walter Scalvini, Senior Software Engineer, explains "Many of our applications were originally written in COBOL. However, we saw need to update older software modules and to develop new applications using up-to-date technologies. Our choice was the Java language and J2EE architecture."

"One important issue that Trend focused on was that any new or updated applications needed to interact seamlessly with the data in existing applications." Scalvini explains.

"Modeling and building new applications must rely on pre-existent data models. There are a lot of databases with hundreds of tables with dozens of columns and relationships. So it was very important to be able to choose a data access architecture for all these data structures."

Andy Grove, CodeFutures' CTO, says "Companies now have massive amounts of data locked in existing databases, often scattered throughout the enterprise. It's essential that companies have a good strategy for realizing the business value locked in their databases and that means a good, future-proof data access architecture."

Trend analyzed its requirements regarding data access architecture, which would be required for all future applications. Scalvini identified the requirements: "Our key requirements were productivity, standardizable approach, quality, consistency, and abstraction."

# CUSTOMER PROFILE



Trend decided to use Data Access Objects for its architecture, because it is a Core J2EE Design Pattern. For Java persistence technology, Trend wishes to remain flexible and choose the most appropriate technology for each solution. Trend carefully examined the Java persistence products available on the market. Scalvini explains the results of the competitive analysis: "After many tests and audits, we chose Firestorm/DAO code generator. Among all the products evaluated, Firestorm/DAO Architect Edition is surely the product that gives us the highest level of freedom, at the same time providing all characteristics we needed".

"It's always a pleasure to win the confidence of a customer, and that's especially true in a competitive situation when the customer is very thorough" says PJ Murray, CEO, CodeFutures Software.

The benefits of FireStorm/DAO identified by Trend were:

### **Productivity**

Writing all necessary code by hand for a typical financial application with very large databases would take months of work, whereas FireStorm/DAO produces instant results.

### **Standards, simplicity, and efficiency**

Code generation based on the Data Access Object core J2EE design pattern, a well defined and accepted architecture.

### **Quality**

Generated software that is high quality, easy to read and modify, and bug free.

### **Consistency**

Class structure, methods, parameters, and naming conventions are automatically managed by code generator using templates.

### **Abstraction**

Flexibility in changing generation options, driven by application needs, and choosing the most appropriate persistence technology (EJB, JDO, Hibernate, JDBC).

### **Contact Details**

Web: [www.codefutures.com](http://www.codefutures.com)

Email: [sales@codefutures.com](mailto:sales@codefutures.com)

CodeFutures Software  
Interlocken Center  
11001 W. 120th Avenue  
Suite 400  
Broomfield, CO 80021  
Fax US: +1 303 484 5134  
Phone US: +1 303 604 9669  
Toll Free US: 866 604 9669

CodeFutures Software  
Dorset House  
Suite 32, Duke St.  
Chelmsford, Essex  
United Kingdom CM1 1TB  
Fax UK: +44 845 280 1775  
Phone UK: +44 777 56 88 081

CodeFutures Software  
1 Parkgate Place  
Parkgate Street  
Dublin 8, Ireland  
Fax Ireland: +353 1 6335084  
Phone Ireland: +353 86 2607139

Copyright © 2005 Code Futures Software Ltd. All warranties on this document are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

CodeFutures™ and FireStorm/DAO™ are trademarks of Code Futures Software, Ltd. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. Other company, product, and service names mentioned in this document may be trademarks or service marks of others.