Do You Still Use a Database?

Klaus Wuestefeld
Prevayler
Av João Gualberto 1673, 4andar
Curitiba – Brazil - 80030-001
klaus@objective.com.br

ABSTRACT

This is the overview of the original System Prevalence layer demonstration at OOPSLA'2003.

Categories and Subject Descriptors

H.2.4 [**Database Management**]: Systems – *concurrency, distributed databases, Object-oriented databases, Query processing, Transaction processing.*

General Terms

Performance, Design, Reliability.

Keywords

Prevayler, Prevalence, Orthogonal Persistence, Replication, Java.

1. INTRODUCTION

DBMSs are the single most hampering force acting on the OO community. While they provide our applications with vital services, such as babies born inside the Matrix, we are pitifully atrophied by the restrictions they impose.

Combining system snapshots and transaction logging, Prevayler [1], the original System Prevalence layer for Java, provides transparent persistence and transparent replication for plain old Java business objects in RAM. It does not require pre or post processing of code. It does not require a special VM. Any Java VM will do.

Prevalent Systems, therefore, are no longer restricted to the data-structures, algorithms and query language provided by a DBMS.

Prevalent systems can process ACID transactions just as fast as a traditional database system. Query performance scalability, on the other hand, is typically 3 to 4 orders of magnitude greater in prevalent systems.

With such performance, basic OO techniques start becoming feasible with persistent objects. Using the observer pattern among thousands of persistent objects or running polymorphic queries on millions of them are a few examples that would be way to slow otherwise, even on OO databases.

The architectural simplicity of any System Prevalence layer makes it an extremely robust alternative to database systems. The implementation of the current Prevayler framework, for example, is less than 400 lines of code long.

2. DEMONSTRATION TOPICS

- Introduction to the ridiculously simple concepts of System Prevalence.

- Presentation of Prevayler [1], the original System Prevalence layer for Java.

- Discussion of transaction and query performance scalability.

- Transparent persistence demonstration with the execution of the demo bank application included in the Prevayler distribution.

- Discussion of the source-code to the plain old Java objects in the application.

- Transaction execution and recovery performance demonstration.

- Automatic rollback demonstration.

- Replication demonstration.

- Open questions.

3. ACKNOWLEDGMENTS

Thanks to all Prevayler users and contributors.

4. REFERENCES

[1] Prevayler Web Site, The. http://www.prevayler.org.

OOPSLA'03, October 26–30, 2003, Anaheim, California, USA.

ACM 1-58113-751-6/03/0010.