Contents

<table>
<thead>
<tr>
<th>Editorial</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>A note from the publisher and editor-in-chief to JOT readers</td>
<td>7</td>
</tr>
</tbody>
</table>

COLUMNS

Strategic Software Engineering

Form over Substance
By John McGregor

Form follows function is a ubiquitous design rule which would seem to support substance over form. It simply sets the stage for development processes that first determine what a product should do and then determines the shape of the architecture. And yet how many projects start with an assumption about which architectural style will be used?

Java at Large

Displaying Updated Stock Quotes
By Douglas Lyon

As a part of the architecture of the program for providing dynamic updates of stock prices, we make use of the observer, mediator and command design patterns. We also make use of the JTable and JTableModel. Finally we show how a multi-threaded update can be performed, using the web as a data source.

Business Objects

Ask Not What Your SOA Can Do For You – Ask What You Can Do For Your SOA!
By Mahesh Dodani

The success of organizational agility is highly dependent on the practitioner, and how effective and efficient they are in leveraging the information and best practices and collaborating with other knowledge

Editor-in-Chief: Richard Wiener rsweeney@acm.org
workers in their network. We can look at how to make knowledge networks successful by considering each of the components that make up a knowledge network.

**Cyber Databases**

**On Flash-Based DBMSs: Issues for Architectural Re-Examination**  
*By Sang-Won Lee and Won Kim*

One technical issue with the flash file system is how to deal efficiently with small random writes, that is writing a small amount of data to blocks of the flash memory. The trouble with the small random writes is that they require the very expensive erase operations. One source of the small random writes is updates to metadata.

**Guest Column**

**Introductory Game Programming Instruction with OOP - What is required, How is it addressed, and Which language wins?**  
*By Lakshmi Prayaga*

Game programming is a phase of game development. This phase includes the choice of the programming language, based on the following features:

- Capability of the software to create game assets (graphics and sound files) or use existing game assets required for the game
- Support for features such as collision detection, AI, and math capabilities of the language to match the requirements specified in the design of the game
- Support to Build a GUI
- Animation
- Portability and extensibility
- Provide for a game loop

**REFEREED ARTICLES**

**A Component Model for Integrating Remote Applications and Services via Web Portals**  
*By Rainer Weinreich, Andreas Wiesauer and Thomas Ziebermayr*

Application integration via web portals is the most widely used and least expensive means for integrating enterprise applications and services. Component-based portals enable the composition of web pages from reusable portal components, where each component represents an independent application or service.
Flexible Language Interoperability
By Torbjörn Ekman, Peter Mechlenborg and Ulrik Pagh Schultz

The mapping from language to virtual machine must support the semantics of the language and is normally designed to produce code that is as efficient as possible. Nevertheless, this same mapping is critical for interoperability with objects implemented in other languages.

A Formal Type System for Java
By M. Debbabi and M. Fourati

We believe that a formal study of Java is very useful to clarify, correct and complete its semantics. It can only be advantageous to consolidate and establish Java security, and make it possible to reason about effective compilation. In our view, a semantics theory for Java is not a luxury, it is a necessity.

Early Safety Analysis: from Use Cases to Component-based Software Development
By Yunja Choi

We propose an easy-to-use but formal approach for early safety analysis in the context of component-based software development and illustrate its application with a case example. Our approach aims at adopting formal safety analysis while maintaining flexibility and consistency throughout the development process.

OUTLOOK

A brief outlook to the next issue