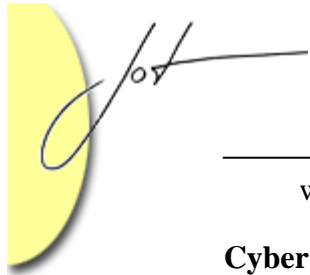


## Contents

	Page
<b>Editorial</b>	5
<hr/>	
<b>COLUMNS</b>	
<hr/>	
<b>Strategic Software Engineering</b>	
<b>Value</b>	7
<i>By John McGregor</i>	
<p>“What we value” and “adding value to a project” may seem to be different uses of the term “value”, but they are synonymous. Whether something is judged to add value depends on what it is we value. One of the jobs of a project manager or product line manager is to align the values of the company, a specific project and its staff.</p>	
<b>Java at Large</b>	
<b>Data Mining Historic Stock Quotes in Java</b>	17
<i>By Douglas Lyon</i>	
<p>Historic stock price data is generally available on the web (using a browser to format the HTML data). Given an HTML data source, we would like to find a way to create an underlying data structure that is type-safe and well formulated.</p>	
<b>The OOP Scene</b>	
<b>Programming the World in a Browser Real Men Don't Do JavaScript Do They?!</b>	25
<i>By Dave Thomas</i>	
<p>The mainstream professional developer community has never taken JavaScript seriously but soon they will have no choice. JavaScript is ready to move to center stage as the development and delivery technology for Web 2.x applications. In the past, most enterprise and product developers flocked to Java or C# while web developers moved to PHP, Perl, Python and more recently Ruby, with most ignoring the</p>	



web based scripting language called JavaScript.

## Cyber Databases

### **On Reflecting Visitors' Opinions Fairly and Accurately on the Web** 31

*By Won Kim, Hyungsuk Ji, and Hyunseung Choo*

The seemingly modest goal of the Web sites' fair and accurate accounting of the visitors' opinions, however, presents significant challenges. The reason is that there are many ways that can lead to distortions of the collective opinions. Some of the ways are attributable to the visitors and others to the site operators. Most Web site operators need to pursue business goals and promote certain points of view. Further, a tiny activist minority may hijack some key policy decisions from the general public, and many Web site visitors do various things to distort the collective opinions.

## Guest Column

### **First Person Shooter Game** 39

*By Rex Cason, Erik Larson, Jonathan Robertson, Jonathan Frisch, George Trice and Dr. Lakshmi Prayaga*

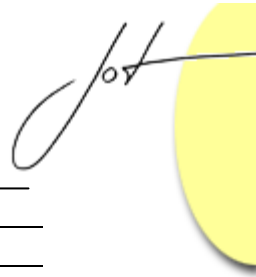
3D game development is an exciting activity for many students. But getting a handle on 3D game development for novices may be a daunting task.

## Guest Column

### **Enterprise Integration in Metadata Environment** 51

*By Raymond Wu*

Enterprise integration minimize the gap between business and IT to improve governance, agility and integrity. Two key processes were used to achieve this objective, the first process relates to how the enterprise goal can be achieved by impacting business process and its service components level, which we call "business semantic" or macro architecture. The second process follows one level below the first one called metadata mediation which called micro process, it constitutes the repository of metadata storing aggregated information and symbolic service information



---

**REFEREED ARTICLES**

---

**Reflective Constraint Management for Languages on Virtual Platforms**

59

*By Mark Royer, Suad Alagic, and Dan Dillon*

Extending an object-oriented type system with assertions makes it possible for programs using reflection to rely on semantic information to ensure correct use of discovered types. Using extended reflective capabilities to access assertions in (dynamically) loaded class objects allows a variety of general and flexible verification techniques. The XVP (Extended Virtual Platform) implements these features by extending the Java Virtual Machine with the proposed functionalities.

**A Typing Scheme for Behavioural Models**

81

*By Ashley McNeile and Nicholas Simons*

Our particular interest has been to provide tools that allow behavioural models to be executed and tested early in the development lifecycle, so that the risk that severe behavioural problems are found at late stages of testing, when rectification can be very expensive, is significantly reduced. The executable models can be viewed as a form of prototype, and the testing and exploration of such prototypes provides a vehicle for users and other stakeholders to engage in and contribute to the modelling process, even if they have no understanding of the notations and concepts used to build the model.

**A Classification of Design Pattern Evolutions**

95

*By Jing Dong, Sheng Yang, and Yongtao Sun*

Designing a software system is hard. Designing a changeable software system is even harder. Design patterns [8] capture expert design experience by partitioning software designs into stable part and changeable part. By separating and encapsulating both parts, the change impact of a software design can be minimized. One of the important goals of design patterns is design for change. Thus, most of design patterns encapsulate future changes that may only affect limited part of a design pattern. This evolution process can be achieved by adding or removing design elements in existing design patterns.

---

**Exceptions in Concurrent Eiffel**

111

*By Phillip J. Brooke and Richard F. Paige*

The Simple Concurrent Object-Oriented Programming (SCOOP) mechanism is proposed as a means to introduce inter-object concurrency into the Eiffel programming language [Mey97, EcI05]. SCOOP extends the Eiffel language by adding one keyword, *separate*, which can be applied to classes, entities and formal routine arguments. Application of *separate* to a class indicates that objects of that class execute in their own conceptual thread of control; application of *separate* to entities (variables) or arguments of routines indicate that these constructs are points of synchronisation.

**Investigating effect of Design Metrics on Fault Proneness in Object-Oriented Systems**

127

*By K.K.Aggarwal, Yogesh Singh, Arvinder Kaur, Ruchika Malhotra*

Demand for quality software has undergone with rapid growth during the last few years. This is leading to an increase in the development of metrics for measuring the properties of software such as coupling, cohesion or inheritance that can be used in early quality assessments. Quality models that explore the relationship between these properties and quality attributes such as fault proneness, maintainability, effort or productivity are needed to use these metrics effectively.

---

**OUTLOOK**

---

**A brief outlook to the next issue**

143