



JOURNAL OF OBJECT TECHNOLOGY

Online at <http://www.jot.fm>. Published by ETH Zurich, Chair of Software Engineering ©JOT, 2006

Vol. 5 No. 3, Special issue: .NET Technologies 2005 Conference, April 2006

Contents

| | Page |
|------------------|------|
| Editorial | 3 |

REFEREED ARTICLES

| | |
|---|---|
| Modeling the .NET CLR Exception Handling Mechanism for a Mathematical Analysis | 5 |
|---|---|

By Nicu G. Fruja and Egon Börger

We provide a mathematical reference model for the exception handling mechanism of the Common Language Runtime (CLR), the virtual machine underlying the interpretation of .NET programs. The model fills some gap in the ECMA standard for CLR and is used to sketch the exception handling related part of a soundness proof for the CLR bytecode verifier.

| | |
|---|----|
| Static Verification of Code Access Security Policy Compliance of .NET Applications | 35 |
|---|----|

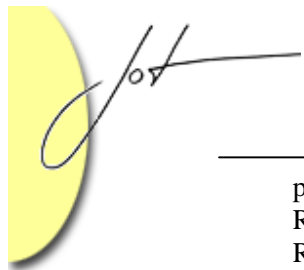
By Jan Smans, Bart Jacobs, and Frank Piessens

In this paper, we propose formal component and method contracts for stack inspection-based sandboxing, and we show that formal verification of these contracts is feasible with state-of-the-art program verification tools. Our contracts are significantly more expressive than existing type systems for stack inspection-based sandboxing. We describe our solution in the context of the sandboxing mechanism in the .NET Framework, called Code Access Security. Our system relies on the Spec# programming language and its accompanying static verification tool.

| | |
|---|----|
| .NET Remoting and Web Services: A Lightweight Bridge between the .NET Compact and Full Framework | 59 |
|---|----|

By Bert Vanhooff, Davy Preuveneers, and Yolande Berbers

Microsoft .NET Compact Framework offers a development platform for mobile applications but is lacking support for .NET Remoting, which is the .NET middleware infrastructure for inter-application communication. The current version of the .NET Compact Framework (1.0, SP2) does support communication using web services. Unfortunately, this support cannot be used in its current form to seamlessly integrate with an existing .NET Remoting application. In this paper, we propose an approach that leverages the present support for web services and augments it to make such integration



possible. Our solution dynamically maps back and forth between .NET Remoting and web service messages without needing to alter the existing Remoting applications.

83

Porting the .NET Compact Framework to Symbian Phones*By Frank Siegemund, Robert Sugar, Alain Gefflaut, and Friedrich van Meigen*

This paper presents our experiences in porting selected parts of the .NET Compact Framework to Symbian smartphones. Our port includes support for basic services such as threading and file access, low-level networking modules as well as Web Services. We also present a portable .NET GUI for the Symbian platform. The paper shows how the programming models of .NET can be efficiently mapped to the runtime structures provided by operating systems for resource-constrained devices such as Symbian OS.

107

An Empirical Study of the Code Pitching Mechanism in the .NET Framework*By David Anthony, Witawas Srisa-an, and Michael Leung*

The .NET Compact Framework achieves fast execution time by compiling methods dynamically instead of using interpretation. While the code cache provides a high-level of reusability, it can also use a large amount of memory. A "code pitching" mechanism can be used to deal with this problem. In this paper, we study the effect of code pitching on the overall performance and memory utilization of .NET applications. We profile the access behavior of the compiled methods. We also experiment with various code cache configurations to perform pitching. We find that programs can operate efficiently with a small code cache without incurring substantial recompilation and execution overheads.

OUTLOOK

A brief outlook to the next issue129
