Capsule Reviews

The Capsule Reviews are intended to provide a short succinct review of each paper in the issue, in order to bring the content to a wider readership. This issue's Capsule Reviews were compiled by Fairouz Kamareddine. Professor Kamareddine is an Associate Editor of *The Computer Journal* and is based in the School of Mathematical and Computer Sciences at Heriot-Watt University, Edinburgh, UK.

An event-driven middleware for mobile context awareness. A. T. S. CHAN, S.-N. CHUANG, J. CAO AND H.-V. LEONG

Middleware technologies facilitate high-level network services and abstract service environments for distributed applications. The success of middleware technologies is largely due to their ability to make service support transparent by hiding the implementation details from the applications. However, this transparency is not so easy to achieve in a mobile environment, since it is hard to completely hide the implementation details from the applications because bandwidths may not always be large, connectivity may not be always on, and error rates may not always be low. The means that in mobile environments, instead of transparency, one needs an awareness of the environment that allows application designers to inspect the execution context and adapt the middleware behaviour. The paper argues that applications operating in such environments are oblivious to the operating context and this often results in premature termination. Instead, the paper proposes the development of a middleware with a degree of transparency, while also allowing the direct awareness of applications of the changes of the operating context so that they can participate in resource allocation and adaptation in response to the dynamic operating environment. In such a context-aware middleware, suitable control mechanisms are needed to allow the active participation of applications. This paper describes the design and implementation of a highly adaptive event-driven mobile middleware. The described system represents a mobile middleware that effectively detects environmental changes and supports wireless adaptation. Initial performance results indicate the ability of the system to sustain a relatively high event-processing rate for small numbers of event resources.

Self-stabilizing mutual exclusion under arbitrary scheduler. A. K. DATTA, M. GRADINARIU AND S. TIXEUIL

Mutual exclusion means that exactly one processor is allowed to execute its critical section at any time, while fairness means that every processor must be able to execute its critical section infinitely often. Self-stabilization is a general technique to design a system that tolerates transient faults. A self-stabilization system converges to a legal configuration in a finite number of steps and remains in a legal state until another fault occurs. When a fault occurs in a self-stabilizing system, one removes, repairs or reinitializes the faulty components and the system returns to a good global state in a short time. Providing self-stabilization in general uniform networks can only be probabilistic. However, a common problem in the uniform probabilistic self-stabilizing, mutual exclusion algorithms designed to work under an unfair distributed scheduler, is that there is no upper bound on the time between two entries into the critical section at a particular processor. This paper answers this question and provides a strong probabilistic stabilizing algorithm for the mutual exclusion problem in an anonymous unidirectional ring of any size running under an unfair distributed scheduler. The paper shows that for a network of size n, the maximum expected stabilization time is $O(n^3)$ under the so-called k-bounded scheduler, and $O(n^2)$ under the so-called synchronous scheduler. Three algorithms for three different schedulers are presented in the paper, together with their correctness proofs and their complexity results.

Multiple error filtering in cyclic systems. G. LATIF-SHABGAHI, J. M. BASS AND S. BENNETT

Failing to ensure software correctness can be catastrophic. The Therac25 radiation therapy machine had mistakes in its software which caused six deaths in the USA and Canada. The importance of avoiding such situations in safety-critical systems led to a wide range of techniques used during the design and implementation of software to reduce the risk of failure. Voting algorithms are a technique used to arbitrate between the various results in fault-tolerant systems. Different voting algorithms exist, some of which produce no result or produce an incorrect output when multiple simultaneous errors occur. This paper considers the problem of voting in multiple error scenarios, and makes explicit the tradeoffs between safety and availability in voting algorithm selection. The paper introduces the so-called predictor voters, which can resolve some multiple error conditions. An implementation of first-order, second-order and third-order predictor voters is given, as well as performance criteria for evaluating the safety and availability of voters. The experimental results show that higher-order predictor voters give higher levels of safety than lower-order ones. Furthermore, these experimental results show that under some circumstances, predictor voters are superior to other types of voters.

Robust data compression: consistency checking in the synchronization of variable length codes. S. PERKINS, D. H. SMITH AND A. RYLEY

When variable length coding is used to represent data compactly, errors in the encoded data can propagate leading to all subsequent data being lost. One solution

is to encode the data in small sections, together with a suitable synchronization scheme. In some cases though, synchronization may lead to slippage where an error causes the number of symbols decoded before resynchronization to differ from the actual number of data symbols that have been encoded. This means that the data, although correctly received, may be misinterpreted. Strong synchronization is a synchronization that avoids slippage. The problem with strong synchronization is that it may be permanently lost during the recovery of synchronization, since an entire cycle of codewords may be jumped. This paper studies how to avoid permanent loss of synchronization when very large sets of data are transmitted or sorted. Various synchronization schemes have been proposed in the literature to avoid the loss of strong synchronization. This paper examines these schemes and studies their susceptibility to errors in environments with high random error rates. Probabilities of losing strong synchronization is studied and different consistency checking tests are given, establishing that for the various schemes studied, those with consistency checking have the best performance.

A cellular automata based reconfigurable architecture for hybrid cryptosystems. H. LI AND C. N. ZHANG
Cryptography is an essential requirement for communication privacy and data security. In practical applications, the two types of cryptography (symmetric-key and public-key) are combined to bring their advantages together in the so-called hybrid cryptosystems. Reconfigurable computing achieves much higher performance than Software, while maintaining a higher level of flexibility than hardware. This paper develops a reconfigurable hybrid-crypto architecture based on Programmable Cellular Automata (PCA). This proposed architecture is modular, simple, has regular interconnections and an easy high-speed VLSI implementation that increases the flexibility of security schemes and reduces the total cost of the hybrid cryptosystems.

Coding techniques for fault-tolerant parallel prefix computations in Abelian groups. C. N. HADJICOSTIS
A fault-tolerant system detects and corrects internal faults. An essential aspect for fault-tolerance is the detection of redundancies in order to distinguish between valid and invalid results and states. The traditional approach to fault-tolerance, the so-called N-modular approach, is inherently expensive and inefficient due to a number of replications. This led many researchers to the introduction of various techniques which help make redundancy more efficient. These techniques have been successfully applied in a variety of settings including arithmetic codes. More involved coding techniques, such as the Algorithm-based fault tolerance (ABFT) techniques, have also been introduced and used in a variety of algorithms. However, both the arithmetic coding and the ABFT techniques face the challenge of introducing redundancy efficiently. This paper develops coding techniques for efficiently protecting parallel prefix in Abelian groups using group homomorphisms. The approach of the author is to homorphically embed the given Abelian group into a larger group and to replace the nodes by other nodes capable of performing this larger group operations. The author shows how the corresponding binary tree architecture allows to detect and correct errors by simply paying attention to the leaf nodes of the tree.

A trading service for COTS components. L. IRIBARNE, J. M. TROYA AND A. VALLECILLO
Nowadays, constructing an application involves the use of prefabricated pieces that could have been developed by different people, at different times, and with different uses in mind. Already, much 'commercial off-the-shelf' (COTS) software has been developed for use in component-based software development (CBSD). However, traditional software development methods are not transferable to CBSD. Moreover, the search and selection processes of COTS components face serious limitations, since for example, their black-box nature hinders the understanding of their internal behaviour, and the traders provide limited information and are not integrated into current methodologies. This paper focuses on the requirements for an effective trading service for COTS components. It describes the features that a trader should have and discusses the shortcomings of current traders. Then, the authors present a trading service that tries to address these current shortcomings. An implementation of an internet-based trader for COTS components is given together with a proposal for documenting components. It is also shown how the proposed trading service can be integrated into a spiral CBSD methodology to give partially automated support for COTS components access and selection processes.

A system for measuring function points from an ER–DFD specification. E. LAMMA, P. MELLO AND F. RIGUZZI
Software metrics enable the application of engineering principles to software development. Various software metrics have been developed. This paper concentrates on the Function Points (FP) metrics which measure the size of a system by measuring its functionality. The main problem of FP metrics is that they are not independent of the person doing the counting. Solutions to this problem have been proposed in the literature. One particular solution suggested simplifying the counting rules so that FP can be automatically counted from early system representations. This paper proposes a system FUN for the automatic counting of FP from a software specification in the form of an Entity Relationship (ER) diagram plus a Data Flow Diagram (DFD). FUN has been successfully tested on a number of case studies. Another contribution of the paper is the translation it gives of the International Function Point User Group (IFPUG) counting rules into a rigorous form.

Achieving bounded and predictable recovery using real-time logging. L. C. SHU, J. A. STANKOVIC AND S. H. SON
In real-time systems, one deals with both variant and invariant data. Data is invariant if its values will not change with time, or if its validity intervals are infinite. It is crucial in real-time database systems (RTDB) to recover critical data within a temporally valid state and predictable time

bounds. If a crash happens, the system must be able to recover within a pre-determined time bound and data must be restored to be temporally valid. This is the main concern of the paper, which presents a logging and recovery technique suitable for RTDB. The paper proposes three principles for fault tolerance in RTDB, where temporal validity and time predictability are the primary design criteria. The proposed approach is able to detect critical data which is indispensable to the system and which must be recovered in case of failure. Moreover, the approach adapts logging strategies to suit the real-time data in use. For non-critical data, only invariant data is logged, whereas the log records for critical variant data are organized according to the last valid time instant stored in each log record. The logging and failure recovery algorithm is provided together with its properties. More importantly, the approach is independent of the technologies used, and the authors exploit the application properties to improve the performance of the recovery algorithm. Performance evaluation of the proposed approach is provided for different system parameters.