Journal of Systems Architecture 111 (2020) 101852

Contents lists available at ScienceDirect

Journal of Systems Architecture

journal homepage: www.elsevier.com/locate/sysarc

Editorial

A new start: Introducing the journal-track proceedings of the 24th Ada-Europe conference on reliable software technologies

Ada-Europe, established in 1988, is a not-for-profit international organization chartered to promote the use and the knowledge of the Ada programming language. Ada-Europe's charter reflects the belief that programming languages – including Ada, of course – project a vision of computing, destined to manifest in specific qualities to be sought by the programmers, and to be featured by the resulting software systems. In the case of Ada (as surely and hopefully it is also for other programming languages), the upheld vision promotes the quest for software qualities that exceed the narrow confines of the user community, and speak of societal benefits – reliability: the capability of deserving trust – which are especially important as most of our professional, social, and even personal activities depend on software services and infrastructures. This premise explains why the most visible of Ada-Europe's public endeavours since 1996 has been the running of an annual international conference on reliable software technologies. As its namesake suggests, this conference was meant to be open and inviting to all, users, practitioners, researchers, who share the goal of producing reliable software. The conference format has always included four principal components: a scientific track, with researchers as protagonists, a technical and experience-based track, with industrial practitioners on the foreground, a tutorial-based educational track, and a technology track, with vendors exhibiting their products and advancements. Since the inception in 1996, for 23 consecutive years, until 2018, the scientific track of the conference had its proceedings in a volume of Springer's Lecture Notes in Computer Science. In the world of Computer Science and Information Engineering, the academic communities of reference for Ada-Europe's conference, timeliness of publication was a paramount value, which made reputed conference proceedings (those with low acceptance ratio and high citation rates) a most desired venue for their scientific products. By the year 2018, the assessment metrics that govern the career progression in the academic system had changed dramatically: the essence of authors' recognition was to be found in high-quartile journal publications. Academic conferences had two ways to compete for authors' products: to attain and hold the reputation of being the single top venue for given topics of interest, or (and) to favour rapid conference-to-journal elevation of the publication material. This was also the course of action taken by Ada-Europe: as of the 2019 edition, its annual conference would have a journal-track component, where the submissions made to it would be routed to a recurrent Special Issue published in an academically-ranked journal, and subject to that journal's review process. Moreover, in order to enable the conference public at large to continue to enjoy the corresponding publications as they previously had via the proceedings volume attached to the conference registration, Ada-Europe made provisions to secure Open Access status to such Special Issue. The virtual Special Issue that this editorial prefaces, is the first product of that change of course. It contains six papers, which form a good representation of the span of arguments addressed by our conference authors, well clear of the "Ada-only topics" prejudice, and testify the healthy variety of authors' provenances for institutions and projects. Allow me to itemize such papers by earlier date of appearance.

- Jorge Real, Sergio Sáez, and Alfons Crespo, all from the Universitad Politècnica de València, Spain, authored "*A hierarchical architecture for time- and event-triggered real-time systems*" (Volume 101, December 2019, 101652). Their work illustrates a programmable architecture for combining the execution of time- and event-triggered real-time task sets. By cleverly playing with Ada's built-in real-time features, the authors' concept and artefact enable users to fine-tune, at a fairly high level of programming, the scheduling decisions that will occur at run time, to warrant reduced jitter and prompt service to non-periodic events. The work includes a Ravenscar implementation of the scheduler and a library of accompanying utilities.
- Xiaotian Dai and Alan Burns, from the University of York, UK, authored "*Period adaptation of real-time control tasks with fixed-priority scheduling in cyber-physical systems*" (Volume 103, February 2020, 101691). The authors' work address long-lived, non-stop cyber-physical systems, the threat that their evolutionary changes may pose to the guarantees of schedulability verified before first deployment, and the opportunity to leverage the knowledge acquired from extended periods of execution to reduce the uncertainties intrinsically present in the temporal specifications of control tasks made in the original system models. This work presents and demonstrate an adaptation method that actively extends the period of control tasks at run-time based on historical measurements, leading to lower power consumption or to the accommodation of increased computation resource demands from other components of the system.
- Sara Royuela and Eduardo Quiñones, from the Barcelona Supercomputing Center, Spain, and Luís Miguel Pinho, from the Polytechnic Institute of Porto, Portugal, authored "*Enabling Ada and OpenMP runtimes interoperability through template-based execution*" (Volume 105, May 2020, 101702). Their work addresses the need to support parallel computation in safety-critical systems, hence with guarantees of safety and robustness. To achieve their goal, the authors propose and demonstrate a way to integrate OpenMP, a de facto standard in high-performance computing, with prospective enhancements to the 202X version of Ada, and show it amenable to static analysis of schedulability.

https://doi.org/10.1016/j.sysarc.2020.101852

Available online 4 August 2020

- Shinhyung Yang, Seongho Jeong, Byunguk Min, Yeonsoo Kim, and Bernd Burgstaller, from the Yonsei University in Korea, together with Johann Blieberger, from TU Wien, Austria, authored "*Design-space evaluation for non-blocking synchronization in Ada: lock elision of protected objects, concurrent objects, and low-level atomics*" (Volume 110, November 2020, 101764). This work address the problem of providing support for non-blocking synchronization in Ada. The authors' concern is that locks, while being very cleverly, efficiently and safely treated in the single-CPU real-time support of the language, prevent the natural scaling of parallel programs on multicore architectures. To remove this obstacle, this work proposes and experimentally evaluates techniques that employ atomic read–modify–write operations in the style of C++11, combined with relaxed memory consistency models.
- Irfan Šljivo and Barbara Gallina, from Mälardalen University, Sweden, Garazi Juez Uriagereka, from Tecnalia, Spain, and Stefano Puri, from Intecs, Italy, authored "*Guiding assurance of architectural design patterns for critical applications*" (Volume 110, November 2020, 101765). This work addresses the general problem of reliable reuse of software in critical systems. More specifically, the authors study how the adoption of certain design patterns aimed to facilitate reuse may be accompanied by specific assurance methods. The paper demonstrates how the information specified in the suite of design pattern templates prototyped by the authors, is capable of yielding an automated instantiation of the assurance argumentation that should accompany each pattern applied to the system.
- Alejandro R. Mosteo, from the Centro Universitario de la Defensa de Zaragoza, Spain, authored "*Reactive programming in Ada 2012 with RxAda*" (Volume 110, November 2020, 101784). This work presents the design of an Ada implementation of ReactiveX, a popular functional reactive programming framework for asynchronous, event-based, multithreaded programming. The author's rationale for this work is that the reactive programming model has properties that fit well the safety and maintainability culture of Ada and therefore make their combination attractive. This paper illustrates the design and implementation of the proposed library, which over the years have matured to the point of having all fundamental building blocks available for the programmer to enjoy the language-agnostic ReactiveX approach without stepping out of the Ada tool chain.

As the curator of the process that led to this publication, I revel in prizing the authors of the papers featured in it, as well as in offering Ada-Europe's conference public and the general readers of this journal, Open Access products of excellent quality.

Tullio Vardanega
University of Padova, Italy
E-mail address: tullio.vardanega@math.unipd.it.