

Programming Support Innovations for Emerging Distributed Applications (PSI EtA— $\psi\eta$)

A SPLASH 2010 workshop summary

Eli Tilevich
Virginia Tech
tilevich@cs.vt.edu

Patrick Eugster
Purdue University
peugster@cs.purdue.edu

Abstract

Distribution has become a necessity for the majority of computing domains, but developing distributed applications remains a highly delicate and complex task. Several emerging distributed computing and application domains, including cloud computing, service-oriented computing, stream processing, sensor networks, and context-aware computing, pose unprecedented challenges to the programmer. Applications in these domains can deliver tangible benefits to the user once they enter the mainstream of industrial software development. What is hindering the wide adoption and use of these applications is the prevalence of ad-hoc programming practices in their software development process. The goal of this workshop is to explore programming support innovations that can address the incongruence between the advanced programming requirements of emerging distributed applications and the current state of the art of their programming support. To that end, the workshop will provide a venue for free and open discussions among academic researchers and industry practitioners of distributed applications.

Categories and Subject Descriptors D.1.3 [*Programming Techniques*]: Concurrent Programming—distributed programming; D.2.11 [*Software Engineering*]: Software Architecture—domain-specific architectures; languages; D.2.13 [*Software Engineering*]: Reusable Software—domain engineering; D.3.2 [*Programming Languages*]: Language Classifications—concurrent, distributed, and parallel languages; extensible language usages; D.3.3 [*Programming Languages*]: Language Construct and Features—modules,

packages; D.3.4 [*Programming Languages*]: Processors—code generation; D.4.7 [*Operating Systems*]: Organization and Design—distributed system; C.2.1 [*Computer-Communication Networks*]: Network Architecture and Design—distributed networks; C.2.4 [*Computer-Communication Networks*]: Distributed Systems—client/server; distributed applications

General Terms Design, Experimentation, Languages

Keywords programming support, distributed computing, service-oriented computing, cloud computing, stream processing, sensor networks, context-aware computing, cyber-physical systems, geo-spatial systems, data-intensive computing, programming frameworks, domain specific languages, code generators, middleware systems, program transformation systems, extensible languages, component technologies, product-line architectures, advanced separation of concerns

1. Main Theme and Goals

This workshop will provide an avenue for researchers and practitioners to explore and address some of the most salient challenges of developing emerging distributed applications.

1.1 Emerging Distributed Applications

Distribution has namely become required for the majority of application domains, and several emerging domains pose unprecedented challenges to the programmer. Examples include: service-oriented computing, cloud computing, stream processing, sensor networks, context-aware computing, cyber-physical systems, geo-spatial systems, data-intensive computing.

Addressing challenges that arise in the development of applications for such emerging distributed settings requires innovation with respect to programming support, including abstractions, techniques, and tools. This workshop aims at filling this gap.

1.2 Contributions Sought

Possible topics for contributions include, but are not limited to: programming frameworks, domain specific languages, code generators, middleware systems, program transformation systems, extensible languages, component technologies, product-line architectures, advanced separation of concerns.

Distributed applications present some of the most salient challenges for researchers and practitioners alike. A large number of top quality recent research publications are concerned with this subject directly or indirectly (e.g., at software engineering conferences ICSE, FSE, at programming languages conferences PLDI, POPL, ECOOP, OOPSLA, and at systems conferences EuroSys, SOSP, NSDI, SIGCOMM). The software industry has introduced numerous commercial applications addressing various challenges of distributed application development, including programming languages, libraries, frameworks, and middleware support. This recent surge of interest in programming support of the emerging distributed applications in these separate communities brings about two insights. Firstly, the challenges of providing adequate programming support are widespread and cut through a wide spectrum of computing technologies. Secondly, truly synergistic approaches may be necessary to address these challenges.

We are convinced that the time is ripe for a cross-domain effort to address the outstanding challenges and to improve the state of the art in programming support for emerging distributed applications. Of particular interest are fundamental approaches which can benefit *several* different distributed application domains, as different settings have common characteristics in addition to diverging ones. The history of programming abstractions has shown that often solutions developed for one domain can be successfully applied to conquer programming challenges of another domain. Therefore, there is great potential benefit in considering a wide array of different distributed application domains during a single workshop. We expect that considering a diverse set of domains from both the practitioner and researcher perspectives will engender unprecedented opportunities for cross-pollination across different fields and communities.

As a specific example, consider sensor networks and context-aware computing. Both research areas may be viewed as two sides of the same coin — one an infrastructure providing contextual information, while the other applications exploiting that information — and may thus have to deal with similar problems and thus share support. Yet, neither implies the other, and it is important to understand the specific characteristics of each. This workshop aims at promoting the collaboration among different communities by providing a forum for exchanging ideas and for presenting cross-domain research. Applying existing approaches and techniques in one domain to another requires that their assumptions, benefits, and fundamental trade-offs be thoroughly understood.

2. Organizers

PSI-EtA is organized by two program committee chairs:

- Eli Tilevich (Virginia Tech)
- Patrick Eugster (Purdue University)

Eli Tilevich is an assistant professor in the Department of Computer Science at Virginia Tech, where he leads the Software Innovations Lab. Its mission is to tame the complexity of developing and maintaining emerging complex computer systems through novel software technologies. Eli's research interests lie on the intersection of systems and software engineering, with a particular emphasis on distributed systems. Some of his recent publications appeared in ECOOP'09, ICDCS'09, OOPSLA'09, Middleware'09, and AOSD'10. In 2010, he is serving as a PC member of ICDCS, ICSM, SPLASH, and the SPLASH Doctoral Symposium. In 2009, he chaired the Second ACM Workshop on Hot Topics in Software Upgrades (HotSWUp'09), which was organized at OOPSLA 2009. Eli is the primary organizer of the workshop and point of contact.

Patrick Eugster is an assistant professor in the Department of Computer Science at Purdue University, and head of the Distributed Programming Group. Patrick's area of expertise lies in the intersection of programming languages and distributed systems. Some of his recent publications in the former area appeared in ESOP'09, PLDI'09, ECOOP'09, or COORDINATION'10, and in the latter area in DSN'09, SRDS'10, Sensys'10, or Middleware'10. Patrick is a recipient of the NSF CAREER award. He served as workshop chair for ECOOP 2008, a conference series with a strong tradition of workshops. He also served on its PC, first in 2006 and last in 2011, as well as on several other conference PCs.

Technical Program Committee

- Gustavo Alonso, *ETH Zürich*
- Godmar Back, *Virginia Tech*
- Shigeru Chiba, *Tokyo Institute of Technology*
- Charles Consel, *INRIA / University of Bordeaux*
- John Field, *IBM Research*
- Buğra Gedik, *IBM Research*
- Jon Howell, *Microsoft Research*
- Hans-Arno Jacobsen, *University of Toronto*
- Christine Julien, *UT Austin*
- Valérie Issarny, *INRIA Paris - Rocquencourt*
- Chandra Krintz, *UCSB*
- Erik Meijer, *Microsoft Research*
- Richard Schantz, *BBN Technologies*
- Liuba Shrira, *Brandeis University*
- Mario Südholt, *Ecole des Mines de Nantes*