

IFIP WG 2.11 on Program Generation

Mission Statement

Aim

Generative approaches have the potential to revolutionize software development as automation and components revolutionized manufacturing. At the same time, the abundance of current research in this area indicates that there is a host of technical problems both at the foundational and engineering levels. As such, the aim of this Working Group of researchers and practitioners is to promote progress in this area.

Scope

The scope of this WG includes the design, analysis, generation, and quality control of generative programs and the programs that they generate.

Specific research themes include (but are not limited to) the following areas:

- Foundations: language design, semantics, type systems, formal methods, multi-stage and multi-level languages, validation and verification.
- Design: models of generative programming, domain engineering, domain analysis and design, system family and product line engineering, model-driven development, separation of concerns, aspect-oriented modeling, feature-oriented modeling.
- Engineering: practices in the context of program generation, such as requirements elicitation and management, software process engineering and management, software maintenance, software estimation and measurement.
- Techniques: meta-programming, staging, templates, inlining, macro expansion, reflection, partial evaluation, intentional programming, stepwise refinement, software reuse, adaptive compilation, runtime code generation, compilation, integration of domain-specific languages, testing.
- Tools: open compilers, extensible programming environments, active libraries, frame processors, program transformation systems, program specializers, aspect weavers, tools for domain modeling.
- Applications: IT infrastructure, finance, telecom, automotive, aerospace, space applications, scientific computing, health, life sciences, manufacturing, government, systems software and middle-ware, embedded and real-time systems, generation of non-code artifacts.

Objectives

- Foster collaboration and interaction between researchers from domain engineering, and on language design, meta-programming techniques, and generative methodologies.
- Demonstrate concrete benefits in specific application areas.
- Develop techniques to assess productivity, reliability, and usability.