

11th Workshop on Language Descriptions, Tools and Applications LDTA 2011



Tool
Challenge

March 26 & 27, 2011 in Saarbrücken, Germany, an ETAPS workshop

<http://ldta.info>

Important Dates:

- Abstract submission:
Dec. 15, 2010
- Full paper submission:
Dec. 22, 2010
- Author notification:
Feb. 1, 2011
- Tool challenge submission:
Mar. 05, 2011
- LDTA Workshop:
Mar. 26-27, 2011 ;

Invited Speaker:

- Rinus Plasmeijer, Radboud
University Nijmegen,
The Netherlands

Program Committee:

- Emilie Balland
- Anya Helene Bagge
- Paulo Borba
- John Boyland
- Claus Brabrand, (co-chair)
brabrand@itu.dk
- Jim Cordy
- Kyung-Goo Doh
- Rob Economopoulos
- Laurie Hendren
- Nigel Horspool
- Roberto Ierusalimsky
- Johan Jeuring
- Shane Markstrum
- Sukyoung Ryu
- João Saraiva
- Sylvain Schmitz
- Sibylle Schupp
- Eli Tilevich
- Eric Van Wyk, (co-chair)
evw@cs.umn.edu
- Eelco Visser

Organizing Committee:

- Emilie Balland
- Rob Economopoulos

LDTA is an application and tool-oriented workshop focused on grammarware — software based on grammars in some form. Grammarware applications are typically language processing applications and traditional examples include parsers, program analyzers, optimizers and translators. A primary focus of LDTA is grammarware that is generated from high-level grammar-centric specifications and thus submissions on parser generation, attribute grammar systems, term/graph rewriting systems, and other grammar-related meta-programming tools, techniques, and formalisms are encouraged.

LDTA is also a forum in which theory is put to the test, in many cases on real-world software engineering challenges. Thus, LDTA also solicits papers on the application of grammarware to areas including, but not limited to, the following:

- program analysis, transformation, generation, and verification,
- implementation of domain specific languages,
- reverse engineering and re-engineering,
- refactoring and other source-to-source transformations,
- language definition and language prototyping, and
- debugging, profiling, IDE support, and testing.

Note that LDTA is a well-established workshop similar to other conferences on (programming) language engineering topics such as SLE and GPCE, but is solely focused on grammarware.

Paper Submission: LDTA solicits papers in the following categories.

- research papers — original research results within the scope of LDTA with a clear motivation, description, analysis, and evaluation. 15 pages.
- short research papers — new ideas that have not been completely fleshed out. As a workshop, LDTA strongly encourages these types of submissions. 6 pages.
- experience report papers — description of the use of a grammarware tool or technique to solve a non-trivial applied problem. 15 pages.
- tool demo papers — discussion of an innovative tool or technique. 10 pages.

The final version of the accepted papers will, pending approval, be published in the ACM Digital Library. Authors of the best papers may be invited to write a journal version for a special issue of Science of Computer Programming. The authors of each submission are required to give a presentation at LDTA 2011 and tool demonstration paper presentations are intended to include a significant live, interactive demonstration. Full paper submission instructions are available at <http://ldta.info>.

Tool Challenge: This year LDTA will also be putting theory, as well as techniques and tools, to the test in a new way — in the LDTA Tool Challenge. Tool developers are invited to participate in the Challenge by developing solutions to a range of language processing tasks over a simple but evolving set of imperative programming languages. Tool challenge participants will present highlights of their solution during a special workshop session. They will also contribute to a joint paper on the proposed solutions to be co-authored by all participants after the workshop. A full description of the Challenge is available at <http://ldta.info>. It includes the set of problems and suggestions for discussing the success of a specific tool or technique.