Workshop Preview of the 3rd Workshop on Parsing Programming Languages (Parsing@SLE 2015)

Loek Cleophas
Umeå University, Sweden
Eindhoven University of Technology, The Netherlands
FASTAR group, Stellenbosch University, South Africa
loek@fastar.org

Ali Afroozeh
CWI, The Netherlands
ali.afroozeh@cwi.nl

Abstract

Parsing@SLE is a workshop on parsing programming languages, now in its third edition, and collocated with SLE and SPLASH 2015. It is held in Pittsburgh, Pennsylvania, USA on October 25h 2015. The goal is to bring together today's experts in the field of parsing, in order to hear about ongoing research, explore open questions and possibly forge new collaborations. Parsing@SLE 2015 will have an invited talk and eight regular talks. We expect to attract participants that have been or are developing theory, techniques and tools in the broad area of parsing non-natural languages such as programming languages.

Categories and Subject Descriptors D.3.4 [*Programming Languages*]: Processors—*parsing*

Keywords Parsing, Programming Languages

1. Theme

For the purpose of this workshop "parsing" is a computation that takes a sequence of characters as input and produces a syntax tree or graph as output. This possibly includes tokenization using regular expressions, deriving trees using context-free grammars, and mapping to abstract syntax trees. The topics may include algorithms, implementation and generation techniques, syntax and semantics of meta formalisms (BNF), etc.

The topics of parsing and parser generation, both in theory and in practice, may be old, yet there are still challenging problems with respect to the construction and maintenance of parsers. Especially in the context of real programming languages there are ample theoretical as well as practical obstacles to be taken. Contemporary parsing challenges are caused by programming language evolution and diversity in the face of new application areas such as IDE construction, reverse engineering, software metrics, domain specific (embedded) languages, etc. What are modular meta-formalisms for parser generation? How to obtain (fast and correct) parsers for both legacy and new languages that require more computational power than context-free grammars and regular expressions can provide? How to use increasing parallelism offered by multi-cores and GPUs in parsers? How to enable the verified construction or prototyping of parsers for languages such as COBOL, C++ and Scala without years of effort? How to parse languages in which white-space is significant (e.g. Python)?

SPLASH Companion'15, October 25–30, 2015, Pittsburgh, PA, USA
ACM. 978-1-4503-3722-9/15/10
http://dx.doi.org/10.1145/2814189.2833209

2. Goal

The workshop, to be held in an informal "retreat" format, is to bring together today's experts in the field of parsing and parser construction for programming languages. Participants will present their currently ongoing (unpublished) work as well as explore the challenges that lie ahead. By bringing the whole community together we hope to create synergy and forge new collaborations.

3. Event Details

Parsing@SLE will be a 1-day workshop, just as with previous editions. There will be an invited talk, followed by eight regular talks, but all with ample time for in depth discussion and explanation, both towards the end of talk slots as well as during breaks. Depending on the number of presentations as well as interest from the participants, towards the end of the day there may be a plenary discussion (like in the 2014 edition) or a number of smaller focus groups for discussion of open problems followed by a short joint session to report back (like in the 2013 edition).

Talks to be presented:

- Invited talk: *Parsing Unpreprocessed C Code - The TypeChef Experience*, Christian Kästner
- *Modular Syntax*, Cyrus Omar
- *GLL parsing for embedded languages*, Anastasiya Ragozina
- *Improving Syntactic Completion*, Luís Eduardo S. Amorim, Guido Wachsmuth, Eelco Visser
- *Fastparse: programmable parsers for the 21st century*, Li Haoyi
- *Operator Precedence for Parser Combinators*, Anastasia Izmaylova
- *Disambiguating Grammars with Tree Automata*, Michael D. Adams
- *Name Resolution Strategies in Variability Realization Languages for Software Product Lines*, Sven Schuster, Christoph Seidl, Ina Schaefer
- *Towards Abstract-Syntax-Preserving Grammar Migrations*, Martijn Dwars, Jeffrey Goderie, Eduardo Amorim, Guido Wachsmuth, Eelco Visser

Parsing@SLE is not a publication venue. We plan to collect minutes and the slides of the presentations and publish them on the SLE website (`www.sleconf.org`) for later reference. We will also add a blog entry on the SLE website about the workshop to report on the presentations and discussions. However we will surely allow authors to keep the slides for themselves and keep some comments off the record, for example pending publication of recent results.

The two previous instances of Parsing@SLE were very well attended. In 2013, when SLE was also co-located with SPLASH, the initial instance of Parsing@SLE had over 40 attendees. The next year, when SLE was co-located with ASE in Sweden it had about 35 attendees. We expect a robust group with similar numbers of attendees this year as well.

4. Organizers

Loek Cleophas (Umeå University, Sweden; Eindhoven University of Technology, The Netherlands; and FASTAR group, Stellenbosch University, South Africa; `loek@fastar.org`**—Primary organizer)** is a researcher at Umeå University and Eindhoven University of Technology and a research associate at Stellenbosch University. He has an extensive background in parsing-related topics, particularly tree acceptance and parsing, and string, regular expression, and tree pattern matching. He also has experience in the development of industrially used domain-specific languages, particularly at Dutch high-tech company ASML. He recently was Computer Science track chair 2015 for the South African internationally-oriented SAICSIT conference, and on the organizing committee for the Conference on Implementation and Application of Automata (CIAA) 2015. He has an MSc (2003, with distinction) and doctorate (2008) in Computer Science and Engineering from Eindhoven University of Technology.

Ali Afroozeh (CWI research institute for mathematics and computer science in the Netherlands; `ali.afroozeh@cwi.nl`**—Contact person)** is a PhD student at CWI working on generalized parsing technology. His research goal is to make generalized top-down parsing techniques practical for parsing programming languages. He is tackling several inter-related challenges such as efficient parser implementation, disambiguation, integration of regular expressions, etc. Ali obtained his MSc in Computer Science and Engineering (with distinction) from Eindhoven University of Technology in 2012.